

Multilevel Monte Carlo methods for applications in finance

Mike Giles and Lukasz Szpruch

Oxford-Man Institute of Quantitative Finance
and Mathematical Institute, University of Oxford

December 7, 2012

Abstract

Since Giles introduced the multilevel Monte Carlo path simulation method [18], there has been rapid development of the technique for a variety of applications in computational finance. This paper surveys the progress so far, highlights the key features in achieving a high rate of multilevel variance convergence, and suggests directions for future research.

1 Introduction

In 2001, Heinrich [28], developed a multilevel Monte Carlo method for parametric integration, in which one is interested in estimating the value of $\mathbb{E}[f(x, \lambda)]$ where x is a finite-dimensional random variable and λ is a parameter. In the simplest case in which λ is a real variable in the range $[0, 1]$, having estimated the value of $\mathbb{E}[f(x, 0)]$ and $\mathbb{E}[f(x, 1)]$, one can use $\frac{1}{2}(f(x, 0) + f(x, 1))$ as a control variate when estimating the value of $\mathbb{E}[f(x, \frac{1}{2})]$, since the variance of $f(x, \frac{1}{2}) - \frac{1}{2}(f(x, 0) + f(x, 1))$ will usually be less than the variance of $f(x, \frac{1}{2})$. This approach can then be applied recursively for other intermediate values of λ , yielding large savings if $f(x, \lambda)$ is sufficiently smooth with respect to λ .

Giles' multilevel Monte Carlo path simulation [18] is both similar and different. There is no parametric integration, and the random variable is infinite-dimensional, corresponding to a Brownian path in the original paper. However, the control variate viewpoint is very similar. A coarse path simulations is used as a control variate for a more refined fine path simulation, but since the exact expectation for the coarse path is not known, this is in turn estimated recursively using even coarser path simulation as control variates. The coarsest path in the multilevel hierarchy may have only one timestep for the entire interval of interest.

A similar two-level strategy was developed slightly earlier by Kebaier [31], and a similar multi-level approach was under development at the same time by Speight [42, 43].

In this review article, we start by introducing the central ideas in multilevel Monte Carlo simulation, and the key theorem from [18] which gives the greatly improved computational cost if a number of conditions are satisfied. The challenge then is to construct numerical methods which satisfy these conditions, and we consider this for a range of computational finance applications.

2 Multilevel Monte Carlo

2.1 Monte Carlo

Monte Carlo simulation has become an essential tool in the pricing of derivatives security and in risk management. In the abstract setting, our goal is to numerically approximate the expected value $\mathbb{E}[Y]$, where $Y = P(X)$ is a functional of a random variable X . In most financial applications we are not able to sample X directly and hence, in order to perform Monte Carlo simulations we approximate X with $X_{\Delta t}$ such that $\mathbb{E}[P(X_{\Delta t})] \rightarrow \mathbb{E}[P(X)]$, when $\Delta t \rightarrow 0$. Using $X_{\Delta t}$ to compute N approximation samples produces the standard Monte Carlo estimate

$$\hat{Y} = \frac{1}{N} \sum_{i=1}^N P(X_{\Delta t}^i),$$

where $X_{\Delta t}^i$ is the numerical approximation to X on the i th sample path and N is the number of independent simulations of X . By standard Monte Carlo results $\hat{Y} \rightarrow \mathbb{E}[Y]$, when $\Delta t \rightarrow 0$ and $N \rightarrow \infty$. In practice we perform Monte Carlo simulation with given $\Delta t > 0$ and finite N producing an error to the approximation of $\mathbb{E}[Y]$. Here we are interested in the mean square error that is

$$MSE \equiv \mathbb{E} \left[(\hat{Y} - \mathbb{E}[Y])^2 \right]$$

Our goal in the design of the Monte Carlo algorithm is to estimate Y with accuracy root-mean-square error ε ($MSE \leq \varepsilon^2$), as efficiently as possible. That is to minimize the computational complexity required to achieve the desired mean square error. For standard Monte Carlo simulations the mean square error can be expressed as

$$\begin{aligned} \mathbb{E} \left[(\hat{Y} - \mathbb{E}[Y])^2 \right] &= \mathbb{E} \left[(\hat{Y} - \mathbb{E}[\hat{Y}] + \mathbb{E}[\hat{Y}] - \mathbb{E}[Y])^2 \right] \\ &= \underbrace{\mathbb{E} \left[(\hat{Y} - \mathbb{E}[\hat{Y}])^2 \right]}_{\text{Monte Carlo variance}} + \underbrace{\left(\mathbb{E}[\hat{Y}] - \mathbb{E}[Y] \right)^2}_{\text{bias of the approximation}}. \end{aligned}$$

The Monte Carlo variance is proportional to $\frac{1}{N}$

$$\mathbb{V}[\hat{Y}] = \frac{1}{N^2} \mathbb{V} \left[\sum_{i=1}^N P(X_{\Delta t}^i) \right] = \frac{1}{N} \mathbb{V}[P(X_{\Delta t})].$$

For both Euler-Maruyama and Milstein approximation $|\mathbb{E}[\hat{Y}] - \mathbb{E}[Y]| = \mathcal{O}(\Delta t)$, typically. Hence the mean square error for standard Monte Carlo is given by

$$\mathbb{E} \left[(\hat{Y} - \mathbb{E}[Y])^2 \right] = \mathcal{O}\left(\frac{1}{N}\right) + \mathcal{O}(\Delta t^2).$$

To ensure the root-mean-square error is proportional to ε , we must have $MSE = \mathcal{O}(\varepsilon^2)$ and therefore $1/N = \mathcal{O}(\varepsilon^2)$ and $\Delta t^2 = \mathcal{O}(\varepsilon^2)$, which means $N = \mathcal{O}(\varepsilon^{-2})$ and $\Delta t = \mathcal{O}(\varepsilon)$. The computational cost of standard Monte Carlo is proportional to the number of paths N multiplied by the cost of generating a path, that is the number of timesteps in each sample path. Therefore, the cost is $C = \mathcal{O}(\varepsilon^{-3})$. In the next section we will show that using MLMC we can reduce the complexity of achieving root mean square error ε to $\mathcal{O}(\varepsilon^{-2})$.

2.2 Multilevel Monte Carlo Theorem

In its most general form, multilevel Monte Carlo (MLMC) simulation uses a number of levels of resolution, $\ell = 0, 1, \dots, L$, with $\ell = 0$ being the coarsest, and $\ell = L$ being the finest. In the context of a SDE simulation, level 0 may have just one timestep for the whole time interval $[0, T]$, whereas level L might have 2^L uniform timesteps $\Delta t_L = 2^{-L}T$.

If P denotes the payoff (or other output functional of interest), and P_ℓ denotes its approximation on level ℓ , then the expected value $\mathbb{E}[P_L]$ on the finest level is equal to the expected value $\mathbb{E}[P_0]$ on the coarsest level plus a sum of corrections which give the difference in expectation between simulations on successive levels,

$$\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{\ell=1}^L \mathbb{E}[P_\ell - P_{\ell-1}]. \quad (1)$$

The idea behind MLMC is to independently estimate each of the expectations on the right-hand side of (1) in a way which minimises the overall variance for a given computational cost. Let Y_0 be an estimator for $\mathbb{E}[P_0]$ using N_0 samples, and let Y_ℓ , $\ell > 0$, be an estimator for $\mathbb{E}[P_\ell - P_{\ell-1}]$ using N_ℓ samples. The simplest estimator is a mean of N_ℓ independent samples, which for $\ell > 0$ is

$$Y_\ell = N_\ell^{-1} \sum_{i=1}^{N_\ell} (P_\ell^i - P_{\ell-1}^i). \quad (2)$$

The key point here is that $P_\ell^i - P_{\ell-1}^i$ should come from two discrete approximations for the same underlying stochastic sample (see [39]), so that on finer

levels of resolution the difference is small (due to strong convergence) and so its variance is also small. Hence very few samples will be required on finer levels to accurately estimate the expected value.

The combined MLMC estimator \hat{Y} is

$$\hat{Y} = \sum_{\ell=0}^L Y_{\ell}.$$

We can observe that

$$\mathbb{E}[Y_{\ell}] = N_{\ell}^{-1} \sum_{i=1}^{N_{\ell}} \mathbb{E}[P_{\ell}^i - P_{\ell-1}^i] = \mathbb{E}[P_{\ell}^i - P_{\ell-1}^i],$$

and

$$\mathbb{E}[\hat{Y}] = \sum_{\ell=0}^L \mathbb{E}[Y_{\ell}] = \mathbb{E}[P_0] + \sum_{\ell=1}^L \mathbb{E}[P_{\ell} - P_{\ell-1}] = \mathbb{E}[P_L].$$

Although we are using different levels with different discretization errors to estimate $\mathbb{E}[P]$, the final accuracy depends on the accuracy of the finest level L .

Here we recall the Theorem from [18] (which is a slight generalisation of the original theorem in [18]) which gives the complexity of MLMC estimation.

Theorem 2.1. *Let P denote a functional of the solution of a stochastic differential equation, and let P_{ℓ} denote the corresponding level ℓ numerical approximation. If there exist independent estimators Y_{ℓ} based on N_{ℓ} Monte Carlo samples, and positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \frac{1}{2} \min(\beta, \gamma)$ and*

$$i) \quad |\mathbb{E}[P_{\ell} - P]| \leq c_1 2^{-\alpha \ell}$$

$$ii) \quad \mathbb{E}[Y_{\ell}] = \begin{cases} \mathbb{E}[P_0], & \ell = 0 \\ \mathbb{E}[P_{\ell} - P_{\ell-1}], & \ell > 0 \end{cases}$$

$$iii) \quad \mathbb{V}[Y_{\ell}] \leq c_2 N_{\ell}^{-1} 2^{-\beta \ell}$$

$$iv) \quad C_{\ell} \leq c_3 N_{\ell} 2^{\gamma \ell}, \text{ where } C_{\ell} \text{ is the computational complexity of } Y_{\ell}$$

then there exists a positive constant c_4 such that for any $\epsilon < e^{-1}$ there are values L and N_{ℓ} for which the multilevel estimator

$$Y = \sum_{\ell=0}^L Y_{\ell},$$

has a mean-square-error with bound

$$MSE \equiv \mathbb{E}[(Y - \mathbb{E}[P])^2] < \epsilon^2$$

with a computational complexity C with bound

$$C \leq \begin{cases} c_4 \epsilon^{-2}, & \beta > \gamma, \\ c_4 \epsilon^{-2} (\log \epsilon)^2, & \beta = \gamma, \\ c_4 \epsilon^{-2-(\gamma-\beta)/\alpha}, & 0 < \beta < \gamma. \end{cases}$$

2.3 Improved MLMC

In the previous section we showed that the key step in MLMC analysis is the estimation of variance $\mathbb{V}[P_\ell^i - P_{\ell-1}^i]$. As it will become more clear in the next section, this is related to the strong convergence results on approximations of SDEs, which differentiates MLMC from standard MC, where we only require a weak error bound for approximations of SDEs.

We will demonstrate that in fact the classical strong convergence may not be necessary for a good MLMC variance. In (2) we have used the same estimator for the payoff P_ℓ on every level ℓ , and therefore (1) is a trivial identity due to the telescoping summation. However, in [17] Giles demonstrated that it can be better to use different estimators for the finer and coarser of the two levels being considered, P_ℓ^f when level ℓ is the finer level, and P_ℓ^c when level ℓ is the coarser level. In this case, we require that

$$\mathbb{E}[P_\ell^f] = \mathbb{E}[P_\ell^c] \quad \text{for } \ell = 1, \dots, L, \quad (3)$$

so that

$$E[P_L^f] = \mathbb{E}[P_0^f] + \sum_{\ell=1}^L \mathbb{E}[P_\ell^f - P_{\ell-1}^c].$$

The MLMC Theorem is still applicable to this modified estimator. The advantage is that it gives the flexibility to construct approximations for which $P_\ell^f - P_{\ell-1}^c$ is much smaller than the original $P_\ell - P_{\ell-1}$, giving a larger value for β , the rate of variance convergence in condition *iii*) in the theorem. In the next sections we demonstrate how suitable choices of P_ℓ^f and P_ℓ^c can dramatically increase the convergence of the variance of the MLMC estimator.

The good choice of estimators, as we shall see, often follows from analysis of the problem under consideration from the distributional point of view. We will demonstrate that methods that had been used previously to improve the weak order of convergence can also improve the order of convergence of the MLMC variance.

2.4 SDEs

First, we consider a general class of d -dimensional SDEs driven by Brownian motion. These are the primary object of studies in mathematical finance. In subsequent sections we demonstrate extensions of MLMC beyond the Brownian setting.

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t \geq 0}, \mathbb{P})$ be a complete probability space with a filtration $\{\mathcal{F}_t\}_{t \geq 0}$ satisfying the usual conditions, and let $w(t)$ be a m -dimensional Brownian motion defined on the probability space. We consider the numerical approximation of SDEs of the form

$$dx(t) = f(x(t)) dt + g(x(t)) dw(t), \quad (4)$$

where $x(t) \in \mathbb{R}^d$ for each $t \geq 0$, $f \in C^2(\mathbb{R}^d, \mathbb{R}^d)$, $g \in C^2(\mathbb{R}^d, \mathbb{R}^{d \times m})$, and for simplicity we assume a fixed initial value $x_0 \in \mathbb{R}^d$. The most prominent example of SDEs in finance is a geometric Brownian motion

$$dx(t) = \alpha x(t) dt + \beta x(t) dw(t),$$

where $\alpha, \beta > 0$. Although, we can solve this equation explicitly it is still worthwhile to approximate its solution numerically in order to judge the performance of the numerical procedure we wish to apply to more complex problems. Another interesting example is the famous Heston stochastic volatility model

$$\begin{cases} ds(t) = rs(t) dt + s(t) \sqrt{v(t)} dw_1(t) \\ dv(t) = \kappa(\theta - v(t)) dt + \sigma \sqrt{v(t)} dw_2(t) \\ dw_1 dw_2 = \rho dt, \end{cases} \quad (5)$$

where $r, \kappa, \theta, \sigma > 0$. In this case we do not know the explicit form of the solution and therefore numerical integration is essential in order to price certain financial derivatives using the Monte Carlo method. At this point we would like to point out that the Heston model (5) does not satisfy standard conditions required for numerical approximations to converge. Nevertheless, in this paper we always assume that coefficients of SDEs (4) are sufficiently smooth. We refer to [32, 35, 44] for an overview of the methods that can be applied when the global Lipschitz condition does not hold. We also refer the reader to [33] for an application of MLMC to the SDEs with additive fractional noise.

2.5 Euler and Milstein discretisations

The simplest approximation of SDEs (4) is an Euler-Maruyama (EM) scheme. Given any step size Δt_ℓ , we define the partition $\mathcal{P}_{\Delta t_\ell} := \{n\Delta t_\ell : n = 0, 1, 2, \dots, 2^\ell\}$ of the time interval $[0, T]$, $2^\ell \Delta t = T > 0$. The EM approximation $X_n^\ell \approx x(n\Delta t_\ell)$ has the form [34]

$$X_{n+1}^\ell = X_n^\ell + f(X_n^\ell) \Delta t_\ell + g(X_n^\ell) \Delta w_{n+1}^\ell, \quad (6)$$

where $\Delta w_{n+1}^\ell = w((n+1)\Delta t_\ell) - w(n\Delta t_\ell)$ and $X_0 = x_0$. Equation (6) is written in a vector form and its i^{th} component reads as

$$X_{i,n+1}^\ell = X_{i,n}^\ell + f_i(X_n^\ell) \Delta t_\ell + \sum_{j=1}^m g_{ij}(X_n^\ell) \Delta w_{j,n+1}^\ell.$$

In the classical Monte Carlo setting we are mainly interested in the weak approximation of SDEs (4). Given a smooth payoff $P : \mathbb{R}^d \rightarrow \mathbb{R}$ we say that $X_{2^\ell}^\ell$ converges to $x(T)$ in a weak sense with order α if

$$|\mathbb{E}[P(x(T))] - \mathbb{E}[P(X_T^\ell)]| = \mathcal{O}(\Delta t_\ell^\alpha).$$

Rate α is required in condition (i) of Theorem 2.1. However, for MLMC condition (iii) of Theorem 2.1 is crucial. We have

$$\mathbb{V}_\ell \equiv \text{Var}(P_\ell - P_{\ell-1}) \leq \mathbb{E}[(P_\ell - P_{\ell-1})^2],$$

and

$$\mathbb{E}[(P_\ell - P_{\ell-1})^2] \leq 2 \mathbb{E}[(P_\ell - P)^2] + 2 \mathbb{E}[(P - P_{\ell-1})^2].$$

For Lipschitz continuous payoffs, $(P(x) - P(y))^2 \leq L \|x - y\|^2$, we then have

$$\mathbb{E}[(P_\ell - P)^2] \leq L \mathbb{E}[\|x(T) - X_T^\ell\|^2].$$

It is clear now, that in order to estimate the variance of the MLMC we need to examine strong convergence property. The classical strong convergence on the finite time interval $[0, T]$ is defined as

$$(\mathbb{E}[\|x(T) - X_T^\ell\|^p])^{1/p} = \mathcal{O}(\Delta t_\ell^\xi), \quad \text{for } p \geq 2.$$

For the EM scheme $\xi = 0.5$. In order to deal with path dependent options we often require measure the error in the supremum norm:

$$\left(\mathbb{E} \left[\sup_{0 \leq n \leq 2^\ell} \|x(n \Delta t_\ell) - X_n^\ell\|^p \right] \right)^{1/p} = \mathcal{O}(\Delta t_\ell^\xi) \quad \text{for } p \geq 2.$$

Even in the case of globally Lipschitz continuous payoff P , the EM does not achieve $\beta = 2\xi > 1$ which is optimal in Theorem (2.1). In order to improve the convergence of the MLMC variance the Milstein approximation $X_n \approx x(n \Delta t_\ell)$ is considered, with i^{th} component of the form [34]

$$\begin{aligned} X_{i,n+1}^\ell = & X_{i,n}^\ell + f_i(X_n^\ell) \Delta t_\ell + \sum_{j=1}^m g_{ij}(X_n^\ell) \Delta w_{j,n+1}^\ell \\ & + \sum_{j,k=1}^m h_{ijk}(X_n^\ell) (\Delta w_{j,n}^\ell \Delta w_{k,n}^\ell - \Omega_{jk} \Delta t_\ell - A_{jk,n}^\ell) \end{aligned} \quad (7)$$

where Ω is the correlation matrix for the driving Brownian paths, and $A_{jk,n}^\ell$ is the Lévy area defined as

$$A_{jk,n}^\ell = \int_{n \Delta t_\ell}^{(n+1) \Delta t_\ell} ((w_j(t) - w_j(n \Delta t_\ell)) dw_k(t) - (w_k(t) - w_k(n \Delta t_\ell)) dw_j(t)).$$

The rate of strong convergence ξ for the Milstein scheme is double the value we have for the EM scheme and therefore the MLMC variance for Lipschitz payoffs converges twice as fast. However, this gain does not come without a price. There is no efficient method to simulate Lévy areas, apart from

dimension 2 [14, 41, 45]. In some applications, the diffusion coefficient $g(x)$ satisfies a commutativity property which gives

$$h_{ijk}(x) = h_{ikj}(x) \quad \text{for all } i, j, k.$$

In that case, because the Lévy areas are anti-symmetric (i.e. $A_{jk,n}^l = -A_{kj,n}^l$), it follows that $h_{ijk}(X_n^\ell) A_{jk,n}^l + h_{ikj}(X_n^\ell) A_{kj,n}^l = 0$ and therefore the terms involving the Lévy areas cancel and so it is not necessary to simulate them. However, this only happens in special cases. Clark & Cameron [9] proved for a particular SDE that it is impossible to achieve a better order of strong convergence than the Euler-Maruyama discretisation when using just the discrete increments of the underlying Brownian motion. The analysis was extended by Müller-Gronbach [38] to general SDEs. As a consequence if we use the standard MLMC method with the Milstein scheme without simulating the Lévy areas the complexity will remain the same as for Euler-Maruyama. Nevertheless, Giles and Szpruch showed in [22] that by constructing a suitable antithetic estimator one can neglect the Lévy areas and still obtain a multi-level correction estimator with a variance which decays at the same rate as the scalar Milstein estimator.

2.6 MLMC algorithm

Here we explain how to implement the Monte Carlo algorithm. Let us recall that the MLMC estimator Y is given by

$$\hat{Y} = \sum_{\ell=0}^L Y_\ell.$$

We aim to minimize the computational cost necessary to achieve desirable accuracy ε . As for standard Monte Carlo we have

$$\mathbb{E}[(Y - \mathbb{E}[P(X)])^2] = \underbrace{\mathbb{E}[(Y - \mathbb{E}[\hat{Y}])^2]}_{\text{Monte Carlo variance}} + \underbrace{(\mathbb{E}[P_L] - \mathbb{E}[P(X)])^2}_{\text{bias of the approximation}}.$$

The variance is given by

$$\mathbb{V}[Y] = \sum_{\ell=0}^L \mathbb{V}[Y_\ell] = \sum_{\ell=0}^L \frac{1}{N_\ell} V_\ell,$$

where $V_\ell = \mathbb{V}[P_\ell - P_{\ell-1}]$. To minimize the variance of Y for fixed computational cost $C = \sum_{\ell=0}^L N_\ell \Delta t_\ell^{-1}$, we can treat N_ℓ as continuous variable and use the Lagrange function to find the minimum of

$$L = \sum_{\ell=0}^L \frac{1}{N_\ell} V_\ell + \lambda \left(\sum_{\ell=0}^L N_\ell \Delta t_\ell^{-1} - C \right).$$

First order conditions shows that $N_\ell = \lambda^{-\frac{1}{2}} \sqrt{V_\ell \Delta t_\ell}$, therefore

$$\mathbb{V}[Y] = \sum_{\ell=0}^L \frac{V_\ell}{N_\ell} = \sum_{\ell=0}^L \frac{\sqrt{\lambda}}{\sqrt{V_\ell \Delta t_\ell}} V_\ell.$$

Since we want $\mathbb{V}[Y] \leq \frac{\varepsilon^2}{2}$ we can show that

$$\lambda^{-\frac{1}{2}} \geq 2\varepsilon^{-2} \sum_{\ell=0}^L \sqrt{V_\ell / \Delta t_\ell},$$

thus the optimal number of samples for level ℓ is

$$N_\ell = \left\lceil 2\varepsilon^{-2} \sqrt{V_\ell \Delta t_\ell} \sum_{\ell=0}^L \sqrt{V_\ell / \Delta t_\ell} \right\rceil. \quad (8)$$

Assuming $\mathcal{O}(\Delta t_\ell)$ weak convergence, the bias of the overall method is equal $c\Delta t_L = cT2^{-L}$. If we want the bias to be proportional to $\frac{\varepsilon}{\sqrt{2}}$ we set

$$L_{\max} = \frac{\log(\varepsilon/(cT\sqrt{2}))^{-1}}{\log 2}.$$

From here we can calculate the overall complexity. We can now outline the algorithm

1. Begin with $L=0$;
2. Calculate the initial estimate of V_L using 100 samples.
3. Determine optimal N_ℓ using (8).
4. Generate additional samples as needed for new N_ℓ .
5. if $L < L_{\max}$ set $L := L + 1$ and go to 2.

Most numerical tests suggests that L_{\max} is not optimal and we can substantially improve MLMC by determining optimal L by looking at bias. For more details see [18].

3 Pricing with MLMC

A key application of MLMC is to compute the expected payoff of financial options. We have demonstrated that for globally Lipschitz European payoffs, convergence of the MLMC variance is determined by the strong rate of convergence of the corresponding numerical scheme. However, in many financial applications payoffs are not smooth or are path-dependent. The aim of this

section is to overview results on mean square convergence rates for Euler–Maruyama and Milstein approximations with more complex payoffs. In the case of EM, the majority of payoffs encountered in practice have been analyzed in Giles et al. [20]. Extension of this analysis to the Milstein scheme is far from obvious. This is due to the fact that Milstein scheme gives an improved rate of convergence on the grid points, but this is insufficient for path dependent options. In many applications the behaviour of the numerical approximation between grid points is crucial. The analysis of Milstein scheme for complex payoffs was carried out in [11]. To understand this problem better, we recall a few facts from the theory of strong convergence of numerical approximations. We can define a piecewise linear interpolation of a numerical approximation within the time interval $[n\Delta t_\ell, (n+1)\Delta t_\ell]$ as

$$X^\ell(t) = X_n^\ell + \lambda_\ell(X_{n+1}^\ell - X_n^\ell), \quad \text{for } t \in [n\Delta t_\ell, (n+1)\Delta t_\ell] \quad (9)$$

where $\lambda_\ell \equiv (t - n\Delta t_\ell)/\Delta t_\ell$. Müller-Gronbach [37] has show that for the Milstein scheme (9) we have

$$\mathbb{E}\left[\sup_{0 \leq t \leq T} \|x(t) - X^\ell(t)\|^p\right] = \mathcal{O}(|\Delta t_\ell \log(\Delta t_\ell)|^{p/2}), \quad p \geq 2, \quad (10)$$

that is the same as for the EM scheme. In order to maintain the strong order of convergence we use Brownian Bridge interpolation rather than basic piecewise linear interpolation:

$$\tilde{X}^\ell(t) = X_n^\ell + \lambda_\ell(X_{n+1}^\ell - X_n^\ell) + g(X_n^\ell) (w(t) - w(n\Delta t_\ell) - \lambda \Delta w_{n+1}^l), \quad (11)$$

for $t \in [n\Delta t_\ell, (n+1)\Delta t_\ell]$. For the Milstein scheme interpolated with Brownian bridges we have [37]

$$\mathbb{E}\left[\sup_{0 \leq t \leq T} \|x(t) - \tilde{X}^\ell(t)\|^p\right] = \mathcal{O}(|\Delta t_\ell \log(\Delta t_\ell)|^p).$$

Clearly $\tilde{X}^\ell(t)$ is not implementable, since in order to construct it, the knowledge of the whole trajectory $(w(t))_{0 \leq t \leq T}$ is required. However, we will demonstrate that combining $\tilde{X}^\ell(t)$ with conditional Monte Carlo techniques can dramatically improve the convergence of the variance of the MLMC estimator. This is due to the fact that for suitable MLMC estimators only distributional knowledge of certain functionals of $(w(t))_{0 \leq t \leq T}$ will be required.

3.1 Euler-Maruyama scheme

In this section we demonstrate how to approximate the most common payoffs using the EM scheme (6).

The Asian option we consider has the payoff

$$P = \left(T^{-1} \int_0^T x(t) dt - K\right)^+.$$

Using the piecewise linear interpolation (9) one can obtain the following approximation

$$P_l \equiv T^{-1} \int_0^T X^\ell(t) dt = T^{-1} \sum_{n=0}^{2^\ell-1} \frac{1}{2} \Delta t_\ell (X_n^\ell + X_{n+1}^\ell),$$

Lookback options have payoffs of the form

$$P = x(T) - \inf_{0 \leq t \leq T} x(t).$$

A numerical approximation to this payoff is

$$P_\ell \equiv X_T^\ell - \inf_{0 \leq n \leq 2^\ell} X_n^\ell.$$

For both of these payoffs it can be proved that $V_\ell = \mathcal{O}(\Delta t_\ell)$ [20].

We now consider a digital option, which pays one unit if the asset at the final time exceeds the fixed strike price K , and pays zero otherwise. Thus, the discontinuous payoff function has the form

$$P = \mathbf{1}_{\{x(T) > K\}},$$

with the corresponding EM value

$$P_\ell \equiv \mathbf{1}_{\{X_T^\ell > K\}}.$$

Assuming boundedness of the density of the solution to (4) in the neighborhood of the strike K , it has been proved in [20] that $V_\ell = \mathcal{O}(\Delta t_\ell^{1/2-\delta})$, for any $\delta > 0$. This result has been tightened by Avikainen [3] who proved that $V_\ell = \mathcal{O}(\Delta t_\ell^{1/2} \log \Delta t_\ell)$.

An up-and-out call gives a European payoff if the asset never exceeds the barrier, B , otherwise it pays zero. So, for the exact solution we have

$$P = (x(T) - K)^+ \mathbf{1}_{\{\sup_{0 \leq t \leq T} x(t) \leq B\}},$$

and for the EM approximation

$$P_\ell \equiv (X_T^\ell - K)^+ \mathbf{1}_{\{\sup_{0 \leq n \leq 2^\ell} X_n^\ell \leq B\}}.$$

A down-and-in call knocks in when the minimum asset price dips below the barrier B , so that

$$P = (x(T) - K)^+ \mathbf{1}_{\{\inf_{0 \leq t \leq T} x(t) \leq B\}},$$

and, accordingly,

$$P_l \equiv (X_T^\ell - K)^+ \mathbf{1}_{\{\inf_{0 \leq n \leq 2^\ell} X_n^\ell \leq B\}}.$$

For both of these barrier options we have $V_\ell = \mathcal{O}(\Delta t_\ell^{1/2-\delta})$, for any $\delta > 0$, assuming that $\inf_{0 \leq t \leq T} x(t)$ and $\sup_{0 \leq t \leq T} x(t)$ have bounded density in the neighborhood of B [20].

As summarised in Table 1, numerical results taken from [17] suggest that all of these results are near-optimal.

option	Euler	
	numerical	analysis
Lipschitz	$\mathcal{O}(\Delta t_\ell)$	$\mathcal{O}(\Delta t_\ell)$
Asian	$\mathcal{O}(\Delta t_\ell)$	$\mathcal{O}(\Delta t_\ell)$
lookback	$\mathcal{O}(\Delta t_\ell)$	$\mathcal{O}(\Delta t_\ell)$
barrier	$\mathcal{O}(\Delta t_\ell^{1/2})$	$\mathcal{O}(\Delta t_\ell^{1/2-\delta})$
digital	$\mathcal{O}(\Delta t_\ell^{1/2})$	$\mathcal{O}(\Delta t_\ell^{1/2} \log \Delta t_\ell)$

Table 1: Orders of convergence for V_ℓ as observed numerically and proved analytically for both Euler discretisations; δ can be any strictly positive constant.

3.2 Milstein scheme

In the scalar case of SDEs (4) (that is with $d = m = 1$) the Milstein scheme has the form

$$X_{n+1}^\ell = X_n^\ell + f(X_n^\ell) \Delta t_\ell + g(X_n^\ell) \Delta w_{n+1}^\ell + g'(X_n^\ell) g(X_n^\ell) ((\Delta w_{n+1}^\ell)^2 - \Delta t_\ell), \quad (12)$$

where $g' \equiv \partial g / \partial x$. The analysis of Lipschitz European payoffs and Asian options with Milstein scheme is analogous to EM scheme and it has been proved in [11] that in both these cases $V_\ell = \mathcal{O}(\Delta t_\ell^2)$.

3.2.1 Lookback options

For clarity of the exposition we will express the fine time-step approximation in terms of the coarse time-step, that is $\mathcal{P}'_{\Delta t_\ell} := \{n \Delta t_{\ell-1} : n = 0, \frac{1}{2}, 1, 1 + \frac{1}{2}, 2, \dots, 2^{\ell-1}\}$. The partition for the coarse approximation is given by $\mathcal{P}_{\Delta t_{\ell-1}} := \{n \Delta t_{\ell-1} : n = 0, 1, 2, \dots, 2^{\ell-1}\}$. Therefore, $X_n^{\ell-1}$ corresponds to X_n^ℓ for $n = 0, 1, 2, \dots, 2^{\ell-1}$.

For pricing lookback options with the EM scheme, as an approximation of the minimum of the process we have simply taken $\min_n X_n^\ell$. This approximation could be improved by taking

$$X_{min}^\ell = \min_n \left(X_n^\ell - \beta^* g(X_n^\ell) \Delta t_\ell^{1/2} \right).$$

Here $\beta^* \approx 0.5826$ is a constant which corrects the $\mathcal{O}(\Delta t_\ell^{1/2})$ leading order error due to the discrete sampling of the path, and thereby restores $\mathcal{O}(\Delta t_\ell)$ weak convergence [6]. However, using this approximation, the difference between the computed minimum values and the fine and coarse paths is $\mathcal{O}(\Delta t_\ell^{1/2})$, and hence the variance V_ℓ is $\mathcal{O}(\Delta t_\ell)$, corresponding to $\beta = 1$. In the previous section, this was acceptable because $\beta = 1$ was the best that could be achieved in general with the Euler path discretisation which was used, but we now aim to achieve an improved convergence rate using the Milstein scheme.

In order to improve the convergence, the Brownian Bridge interpolant $\tilde{X}^\ell(t)$ defined in (11) is used. We have

$$\begin{aligned}\min_{0 \leq t < T} \tilde{X}^\ell(t) &= \min_{0 \leq n \leq 2^{\ell-1} - \frac{1}{2}} \left[\min_{n\Delta t_{\ell-1} \leq t < (n+\frac{1}{2})\Delta t_{\ell-1}} \tilde{X}^\ell(t) \right] \\ &= \min_{0 \leq n \leq 2^{\ell-1} - \frac{1}{2}} X_{n,min}^\ell,\end{aligned}$$

where minimum of the fine approximation over the first half of the coarse time-step is given by [24]

$$X_{n,min}^\ell = \frac{1}{2} \left(X_n^\ell + X_{n+\frac{1}{2}}^\ell - \sqrt{\left(X_{n+\frac{1}{2}}^\ell - X_n^\ell \right)^2 - 2g(X_n^\ell)^2 \Delta t_\ell \log U_n^\ell} \right), \quad (13)$$

and minimum of the fine approximation over the second half of the coarse time-step is given by

$$X_{n+\frac{1}{2},min}^\ell = \frac{1}{2} \left(X_{n+\frac{1}{2}}^\ell + X_{n+1}^\ell - \sqrt{\left(X_{n+1}^\ell - X_{n+\frac{1}{2}}^\ell \right)^2 - 2g(X_{n+\frac{1}{2}}^\ell)^2 \Delta t_\ell \log U_{n+\frac{1}{2}}^\ell} \right), \quad (14)$$

where $U_n^\ell, U_{n+\frac{1}{2}}^\ell$ are uniform random variables on the unit interval. For the coarse path, in order to improve the MLMC variance a slightly different estimator is used, see (3). Using the same Brownian increments as we used on the fine path (to guarantee that we stay on the same path), equation (11) is used to define $\tilde{X}_{n+\frac{1}{2}}^{\ell-1} \equiv \tilde{X}^{\ell-1}((n+\frac{1}{2})\Delta t_{\ell-1})$. Given this interpolated value, the minimum value over the interval $[n\Delta t_{\ell-1}, (n+1)\Delta t_{\ell-1}]$ can then be taken to be the smaller of the minima for the two intervals $[n\Delta t_{\ell-1}, (n+\frac{1}{2})\Delta t_{\ell-1}]$ and $[(n+\frac{1}{2})\Delta t_{\ell-1}, (n+1)\Delta t_{\ell-1}]$,

$$\begin{aligned}X_{n,min}^{\ell-1} &= \frac{1}{2} \left(X_n^{\ell-1} + \tilde{X}_{n+\frac{1}{2}}^{\ell-1} - \sqrt{\left(\tilde{X}_{n+\frac{1}{2}}^{\ell-1} - X_n^{\ell-1} \right)^2 - 2(g(X_n^{\ell-1}))^2 \frac{\Delta t_{\ell-1}}{2} \log U_n^\ell} \right), \\ X_{n+\frac{1}{2},min}^{\ell-1} &= \frac{1}{2} \left(\tilde{X}_{n+\frac{1}{2}}^{\ell-1} + X_{n+1}^{\ell-1} - \sqrt{\left(X_{n+1}^{\ell-1} - \tilde{X}_{n+\frac{1}{2}}^{\ell-1} \right)^2 - 2(g(X_n^{\ell-1}))^2 \frac{\Delta t_{\ell-1}}{2} \log U_{n+\frac{1}{2}}^\ell} \right).\end{aligned} \quad (15)$$

Note that $g(X_n^{\ell-1})$ is used for both time steps. It is because we used the Brownian Bridge with diffusion term $g(X_n^{\ell-1})$ to derive both minima. If we changed $g(X_n^{\ell-1})$ to $g(\tilde{X}_{n+\frac{1}{2}}^{\ell-1})$ in $X_{n+\frac{1}{2},min}^{\ell-1}$, this would mean that different Brownian Bridges were used on the first and second half of the coarse time-step and as a consequence condition (3) would be violated. Note also the re-use of the same uniform random numbers U_n^ℓ and $U_{n+\frac{1}{2}}^\ell$ used to compute the fine path minimum. The $\min(X_{n,min}^{\ell-1}, X_{n+\frac{1}{2},min}^{\ell-1})$ has exactly the same distribution as $X_{n,min}^{\ell-1}$, since they are both based on the same Brownian interpolation, and therefore equality (3) is satisfied. Giles et al. [11] proved the following Theorem:

Theorem 3.1. *The multilevel approximation for a lookback option which is a uniform Lipschitz function of $x(T)$ and $\inf_{[0,T]} x(t)$ has $V_l = o(\Delta t_l^{2-\delta})$ for any $\delta > 0$.*

3.3 Conditional Monte Carlo

Giles [17] and Giles et al. [11] have shown that combining conditional Monte Carlo with MLMC results in superior estimators for various financial payoffs.

To obtain an improvement in the convergence of the MLMC variance barrier and digital options, conditional Monte Carlo methods is employed. We briefly describe it here. Our goal is to calculate $\mathbb{E}[P]$. Instead, we can write

$$\mathbb{E}[P] = \mathbb{E}[\mathbb{E}[P \mid Z]],$$

where Z is a random vector. Hence $\mathbb{E}[P \mid Z]$ is an unbiased estimator of $\mathbb{E}[P]$. We also have

$$\text{Var}[P] = \mathbb{E}[\text{Var}[P \mid Z]] + \text{Var}[\mathbb{E}[P \mid Z]],$$

hence $\text{Var}[\mathbb{E}[P \mid Z]] \leq \text{Var}(P)$. In the context of MLMC we obtain a better variance convergence if we condition on different vectors on the fine and the coarse level. That is on the fine level we take $\mathbb{E}[P^f \mid Z^f]$, where $Z^f = \{X_n^\ell\}_{0 \leq n \leq 2^\ell}$. On the coarse level instead of taking $\mathbb{E}[P^c \mid Z^c]$ with $Z^c = \{X_n^{\ell-1}\}_{0 \leq n \leq 2^{\ell-1}}$, we take $\mathbb{E}[P^c \mid Z^c, \tilde{Z}^c]$, where $\tilde{Z}^c = \{\tilde{X}_{n+\frac{1}{2}}^{\ell-1}\}_{0 \leq n \leq 2^{\ell-1}}$ are obtained from equation (11). Condition (3) trivially holds by tower property of conditional expectation

$$\mathbb{E}[\mathbb{E}[P^c \mid Z^c]] = \mathbb{E}[P^c] = \mathbb{E}[\mathbb{E}[P^c \mid Z^c, \tilde{Z}^c]].$$

3.4 Barrier options

The barrier option which is considered is a down-and-out option for which the payoff is a Lipschitz function of the value of the underlying at maturity, provided the underlying has never dropped below a value $B \in \mathbb{R}$,

$$P = f(x(T)) \mathbf{1}_{\{\tau > T\}}.$$

The crossing time τ is defined as

$$\tau = \inf_t \{x(t) < B\}.$$

This requires the simulation of $(x(T), \mathbf{1}_{\{\tau > T\}})$. The simplest method sets

$$\tau^{\Delta t_\ell} = \inf_n \{X_n^\ell < B\}$$

and as an approximation takes $(X_{2^{\ell-1}}^\ell, \mathbf{1}_{\{\tau^{\Delta t_\ell} > 2^{\ell-1}\}})$. But even if we could simulate the process $\{x(n\Delta t_\ell)\}_{0 \leq n \leq 2^{\ell-1}}$ it is possible for $\{x(t)\}_{0 \leq t \leq T}$ to cross

the barrier between grid points. Using the Brownian Bridge interpolation we can approximate $\mathbf{1}_{\{\tau > T\}}$ by

$$\prod_{n=0}^{2^{\ell-1}-\frac{1}{2}} \mathbf{1}_{\{X_{n,min}^{\ell} \geq B\}}.$$

This suggests following the lookback approximation in computing the minimum of both the fine and coarse paths. However, the variance would be larger in this case because the payoff is a discontinuous function of the minimum. A better treatment, which is the one used in [16], is to use the conditional Monte Carlo approach to further smooth the payoff. Since the process X_n^{ℓ} is Markovian we have

$$\begin{aligned} & \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell}) \prod_{n=0}^{2^{\ell-1}-\frac{1}{2}} \mathbf{1}_{\{X_{n,min}^{\ell} \geq B\}}\right] \\ &= \mathbb{E}\left[\mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell}) \prod_{n=0}^{2^{\ell-1}-\frac{1}{2}} \mathbf{1}_{\{X_{n,min}^{\ell} \geq B\}} \mid X_0^{\ell}, \dots, X_{2^{\ell-1}}^{\ell}\right]\right] \\ &= \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell}) \prod_{n=0}^{2^{\ell-1}-\frac{1}{2}} \mathbb{E}\left[\mathbf{1}_{\{X_{n,min}^{\ell} \geq B\}} \mid X_n^{\ell}, X_{n+1}^{\ell}\right]\right] \\ &= \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell}) \prod_{n=0}^{2^{\ell-1}-\frac{1}{2}} (1 - p_n^{\ell})\right], \end{aligned}$$

where from [24]

$$\begin{aligned} p_n^{\ell} &= \mathbb{P}\left(\inf_{n\Delta t_{\ell} \leq t < (n+\frac{1}{2})\Delta t_{\ell}} \tilde{X}(t) < B \mid X_n^{\ell}, X_{n+\frac{1}{2}}^{\ell}\right) \\ &= \exp\left(\frac{-2(X_n^{\ell} - B)^+(X_{n+\frac{1}{2}}^{\ell} - B)^+}{g(X_n^{\ell})^2 \Delta t_{\ell}}\right), \end{aligned}$$

and

$$\begin{aligned} p_{n+\frac{1}{2}}^{\ell} &= \mathbb{P}\left(\inf_{(n+\frac{1}{2})\Delta t_{\ell} \leq t < (n+1)\Delta t_{\ell}} \tilde{X}(t) < B \mid X_{n+\frac{1}{2}}^{\ell}, X_{n+1}^{\ell}\right) \\ &= \exp\left(\frac{-2(X_{n+\frac{1}{2}}^{\ell} - B)^+(X_{n+1}^{\ell} - B)^+}{g(X_{n+\frac{1}{2}}^{\ell})^2 \Delta t_{\ell}}\right). \end{aligned}$$

Hence, for the fine path this gives

$$P_{\ell}^f = f(X_{2^{\ell-1}}^{\ell}) \prod_{n=0}^{2^{\ell-2}-\frac{1}{2}} (1 - p_n^{\ell}), \quad (16)$$

The payoff for the coarse path is defined similarly. However, in order to reduce the variance, we subsample $\tilde{X}_{n+\frac{1}{2}}^{\ell-1}$, as we did for lookback options, from the Brownian Bridge connecting $X_n^{\ell-1}$ and $X_{n+1}^{\ell-1}$

$$\begin{aligned}
& \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell-1}) \prod_{n=0}^{2^{\ell-1}-1} \mathbf{1}_{\{X_{n,min}^{\ell-1} \geq B\}}\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell-1}) \prod_{n=0}^{2^{\ell-1}-1} \mathbf{1}_{\{X_{n,min}^{\ell-1} \geq B\}} \mid X_0^{\ell-1}, \tilde{X}_{\frac{1}{2}}^{\ell-1}, \dots, \tilde{X}_{2^{\ell-1}-\frac{1}{2}}^{\ell-1}, X_{2^{\ell-1}}^{\ell-1}\right]\right] \\
&= \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell-1}) \prod_{n=0}^{2^{\ell-1}-1} \mathbb{E}\left[\mathbf{1}_{\{X_{n,min}^{\ell-1} \geq B\}} \mid X_n^{\ell-1}, \tilde{X}_{n+\frac{1}{2}}^{\ell-1}, X_{n+1}^{\ell-1}\right]\right] \\
&= \mathbb{E}\left[f(X_{2^{\ell-1}}^{\ell-1}) \prod_{n=0}^{2^{\ell-1}-1} (1 - p_{1,n}^{\ell-1})(1 - p_{2,n}^{\ell-1})\right],
\end{aligned}$$

where

$$p_{1,n}^{\ell-1} = \exp\left(\frac{-2(X_n^{\ell-1} - B)^+(\tilde{X}_{n+\frac{1}{2}}^{\ell-1} - B)^+}{g(X_n^{\ell-1})^2 \Delta t_\ell}\right),$$

and

$$p_{2,n}^{\ell-1} = \exp\left(\frac{-2(\tilde{X}_{n+\frac{1}{2}}^{\ell-1} - B)^+(X_{n+1}^{\ell-1} - B)^+}{g(X_n^{\ell-1})^2 \Delta t_l}\right).$$

Note that the same $g(X_n^{\ell-1})$ is used (rather than using $g(\tilde{X}_{n+\frac{1}{2}}^{\ell-1})$ in $p_{2,n}^{\ell-1}$) to calculate both probabilities for the same reason as we did for lookback options. The final estimator can be written as

$$P_{\ell-1}^c = f(X_{2^{\ell-1}}^{\ell-1}) \prod_{n=0}^{2^{\ell-1}-1} (1 - p_{1,n}^{\ell-1})(1 - p_{2,n}^{\ell-1}). \quad (17)$$

Giles et al. [11] proved the following theorem

Theorem 3.2. *Provided $\inf_{[0,T]} |g(B)| > 0$, and $\inf_{[0,T]} x(t)$ has a bounded density in the neighbourhood of B , then the multilevel estimator for a down-and-out barrier option has variance $V_\ell = o(\Delta t_\ell^{3/2-\delta})$ for any $\delta > 0$.*

The reason the variance is approximately $o(\Delta t_\ell^{3/2-\delta})$ instead of $\mathcal{O}(\Delta t_\ell^2)$ is the following: due to the strong convergence property the probability of the numerical approximation being outside $\Delta t_\ell^{1-\delta}$ -neighbourhood of the solution to the SDE (4) is arbitrary small, that is for any $\varepsilon > 0$

$$\begin{aligned}
& \mathbb{P}\left(\sup_{0 \leq n \Delta t_\ell \leq T} \|x(n \Delta t_\ell) - X_n^\ell\| \geq \Delta t_\ell^{1-\varepsilon}\right) \\
& \leq \Delta t_\ell^{-p+p\varepsilon} \mathbb{E}\left[\sup_{0 \leq n \Delta t_\ell \leq T} \|x(n \Delta t_\ell) - X_n^\ell\|^p\right] = \mathcal{O}(\Delta t_\ell^{p\varepsilon}).
\end{aligned} \quad (18)$$

If $\inf_{[0,T]} x(t)$ is outside the $\Delta t_\ell^{1/2}$ -neighborhood of the barrier B then by (18) it is shown that so are numerical approximations. The probabilities of crossing the barrier in that case are asymptotically either 0 or 1 and essentially we are in the Lipschitz payoff case. If the $\inf_{[0,T]} x(t)$ is within the $\Delta t_\ell^{1/2}$ -neighbourhood of the barrier B then so are the numerical approximations. In that case it can be shown that $\mathbb{E}[(P_\ell^f - P_{\ell-1}^c)^2] = \mathcal{O}(\Delta t^{1-\delta})$ but due to the bounded density assumption, the probability that $\inf_{[0,T]} x(t)$ is within $\Delta t_\ell^{1/2}$ -neighbourhood of the barrier B is of order $\Delta t_\ell^{1/2-\delta}$. Therefore the overall MLMC variance is $V_\ell = o(\Delta_\ell^{3/2-\delta})$ for any $\delta > 0$.

3.5 Digital options

A digital option has a payoff which is a discontinuous function of the value of the underlying asset at maturity, the simplest example being

$$P = \mathbf{1}_{\{x(T) > B\}}.$$

Approximating $\mathbf{1}_{\{x(T) > B\}}$ based only on simulations of $x(T)$ by Milstein scheme will lead to an $\mathcal{O}(\Delta t_\ell)$ fraction of the paths having coarse and fine path approximations to $x(T)$ on either side of the strike, producing $P_\ell - P_{\ell-1} = \pm 1$, resulting in $V_\ell = \mathcal{O}(\Delta t_\ell)$. To improve the variance to $\mathcal{O}(\Delta t_\ell^{3/2-\delta})$ for all $\delta > 0$, the conditional Monte Carlo method is used to smooth the payoff (see section 7.2.3 in [24]). This approach was proved to be successful in Giles et al. [11] and was tested numerically in [16],

If $X_{2^{\ell-1}-\frac{1}{2}}^\ell$ denotes the value of the fine path approximation one time-step before maturity, then the motion thereafter is approximated as Brownian motion with constant drift $f(X_{2^{\ell-1}-\frac{1}{2}}^\ell)$ and volatility $g(X_{2^{\ell-1}-\frac{1}{2}}^\ell)$. The conditional expectation for the payoff is the probability that $X_{2^{\ell-1}}^\ell > B$ after one further time-step, which is

$$P_\ell^f = \mathbb{E} \left[\mathbf{1}_{\{X_{2^{\ell-1}}^\ell > B\}} \mid X_{2^{\ell-1}-\frac{1}{2}}^\ell \right] = \Phi \left(\frac{X_{2^{\ell-1}-\frac{1}{2}}^\ell + f(X_{2^{\ell-1}-\frac{1}{2}}^\ell) \Delta t_\ell - B}{|g(X_{2^{\ell-1}-\frac{1}{2}}^\ell)| \sqrt{\Delta t_\ell}} \right), \quad (19)$$

where Φ is the cumulative Normal distribution.

For the coarse path, we note that given the Brownian increment $\Delta w_{2^{\ell-1}-\frac{1}{2}}^{\ell-1}$ for the first half of the last coarse time-step (which comes from the fine path simulation), the probability that $X_{2^{\ell-1}}^\ell > B$ is

$$\begin{aligned} P_{\ell-1}^c &= \mathbb{E} \left[\mathbf{1}_{\{X_{2^{\ell-1}}^{\ell-1} > B\}} \mid X_{2^{\ell-1}-1}^{\ell-1}, \Delta w_{2^{\ell-1}-\frac{1}{2}}^{\ell-1} \right] \\ &= \Phi \left(\frac{X_{2^{\ell-1}-1}^{\ell-1} + f(X_{2^{\ell-1}-1}^{\ell-1}) \Delta t_{\ell-1} + g(X_{2^{\ell-1}-1}^{\ell-1}) \Delta w_{2^{\ell-1}-\frac{1}{2}}^{\ell-1} - B}{|g(X_{2^{\ell-1}-1}^{\ell-1})| \sqrt{\Delta t_\ell}} \right). \end{aligned} \quad (20)$$

The conditional expectation of (20) is equal to the conditional expectation of $P_{\ell-1}^f$ defined by (19) on level $\ell-1$, and so equality (3) is satisfied. A bound on the variance of the multilevel estimator is given by the following result:

Theorem 3.3. *Provided $g(B) \neq 0$, and $x(t)$ has a bounded density in the neighbourhood of B , then the multilevel estimator for a digital option has variance $V_l = o(\Delta t_l^{3/2-\delta})$ for any $\delta > 0$.*

Results of the above section were tested numerically in [17] and are summarized in the table 2.

option	Milstein	
	numerical	analysis
Lipschitz	$\mathcal{O}(\Delta t_\ell^2)$	$\mathcal{O}(\Delta t_\ell^2)$
Asian	$\mathcal{O}(\Delta t_\ell^2)$	$\mathcal{O}(\Delta t_\ell^2)$
lookback	$\mathcal{O}(\Delta t_\ell^2)$	$o(\Delta t_\ell^{2-\delta})$
barrier	$\mathcal{O}(\Delta t_\ell^{3/2})$	$o(\Delta t_\ell^{3/2-\delta})$
digital	$\mathcal{O}(\Delta t_\ell^{3/2})$	$o(\Delta t_\ell^{3/2-\delta})$

Table 2: Orders of convergence for V_l as observed numerically and proved analytically for Milstein discretisations; δ can be any strictly positive constant.

4 Greeks with MLMC

Accurate calculation of prices is only one objective of Monte Carlo simulations. Even more important in some ways is the calculation of the sensitivities of the prices to various input parameters. These sensitivities, known collectively as the “Greeks”, are important for risk analysis and mitigation through hedging.

Here we follow the results by Burgos et al. [8] to present how MLMC can be applied in this setting. The pathwise sensitivity approach (also known as Infinitesimal Perturbation Analysis) is one of the standard techniques for computing these sensitivities [24]. However, the pathwise approach is not applicable when the financial payoff function is discontinuous. One solution to these problems is to use the Likelihood Ratio Method (LRM) but its weaknesses are that the variance of the resulting estimator is usually $\mathcal{O}(\Delta t_l^{-1})$.

Three techniques are presented that improve MLMC variance: payoff smoothing using conditional expectations [24]; an approximation of the above technique using path splitting for the final timestep [2]; the use of a hybrid combination of pathwise sensitivity and the Likelihood Ratio Method [19]. We discuss the strengths and weaknesses of these alternatives in different multi-level Monte Carlo settings.

4.1 Monte Carlo Greeks

Consider the approximate solution of the general SDE (4) using Euler discretisation (6). The Brownian increments can be defined to be a linear transformation of a vector of independent unit Normal random variables Z .

The goal is to efficiently estimate the expected value of some financial payoff function $P(x(T))$, and numerous first order sensitivities of this value with respect to different input parameters such as the volatility or one component of the initial data $x(0)$. In more general cases P might also depend on the values of process $\{x(t)\}_{0 \leq t \leq T}$ at intermediate times.

The pathwise sensitivity approach can be viewed as starting with the expectation expressed as an integral with respect to Z :

$$V_\ell \equiv \mathbb{E} [P(X_n^\ell(Z, \theta))] = \int P(X_n^\ell(Z, \theta)) p_Z(Z) dZ. \quad (21)$$

Here θ represents a generic input parameter, and the probability density function for Z is

$$p_Z(Z) = (2\pi)^{-d/2} \exp(-\|Z\|_2^2/2),$$

where d is the dimension of the vector Z .

Let $X_n^\ell = X_n^\ell(Z, \theta)$. If the drift, volatility and payoff functions are all differentiable, (21) may be differentiated to give

$$\frac{\partial V_\ell}{\partial \theta} = \int \frac{\partial P(X_n^\ell)}{\partial X_n^\ell} \frac{\partial X_n^\ell}{\partial \theta} p_Z(Z) dZ, \quad (22)$$

with $\frac{\partial X_n^\ell}{\partial \theta}$ being obtained by differentiating (6) to obtain

$$\begin{aligned} \frac{\partial X_{n+1}^\ell}{\partial \theta} &= \frac{\partial X_n^\ell}{\partial \theta} + \left(\frac{\partial f(X_n^\ell, \theta)}{\partial X_n^\ell} \frac{\partial X_n^\ell}{\partial \theta} + \frac{\partial f(X_n^\ell, \theta)}{\partial \theta} \right) \Delta t_l \\ &\quad + \left(\frac{\partial g(X_n^\ell, \theta)}{\partial X_n^\ell} \frac{\partial X_n^\ell}{\partial \theta} + \frac{\partial g(X_n^\ell, \theta)}{\partial \theta} \right) \Delta w_{n+1}^l. \end{aligned} \quad (23)$$

We assume that $Z \rightarrow \Delta w_{n+1}^l$ mapping does not depend on θ . It can be proved that (22) remains valid (that is we can interchange integration and differentiation) when the payoff function is continuous and piecewise differentiable, and the numerical estimate obtained by standard Monte Carlo with M independent path simulations

$$M^{-1} \sum_{m=1}^M \frac{\partial P(X_n^{\ell, m})}{\partial X_n^\ell} \frac{\partial X_n^{\ell, m}}{\partial \theta}$$

is an unbiased estimate for $\partial V/\partial \theta$ with a variance which is $\mathcal{O}(M^{-1})$, if $P(x)$ is Lipschitz and the drift and volatility functions satisfy the standard conditions [34].

Performing a change of variables, the expectation can also be expressed as

$$V_\ell \equiv \mathbb{E} [P(X_n^\ell)] = \int P(x) p_{X_n^\ell}(x, \theta) dx, \quad (24)$$

where $p_{X_n^\ell}(x, \theta)$ is the probability density function for X_n^ℓ which will depend on all of the inputs parameters. Since probability density functions are usually

smooth, (24) can be differentiated to give

$$\frac{\partial V_\ell}{\partial \theta} = \int P(x) \frac{\partial p_{X_n^\ell}}{\partial \theta} dx = \int P(x) \frac{\partial(\log p_{X_n^\ell})}{\partial \theta} p_{X_n^\ell} dx = \mathbb{E} \left[P(x) \frac{\partial(\log p_{X_n^\ell})}{\partial \theta} \right].$$

which can be estimated using the unbiased Monte Carlo estimator

$$M^{-1} \sum_{m=1}^M P(X_n^{\ell,m}) \frac{\partial \log p_{X_n^\ell}(X_n^{\ell,m})}{\partial \theta}$$

This is the Likelihood Ratio Method. Its great advantage is that it does not require the differentiation of $P(X_n^\ell)$. This makes it applicable to cases in which the payoff is discontinuous, and it also simplifies the practical implementation because banks often have complicated flexible procedures through which traders specify payoffs. However, it does have a number of limitations, one being a requirement of absolute continuity which is not satisfied in a few important applications such as the LIBOR market model [24].

4.2 Multilevel Monte Carlo Greeks

The MLMC method for calculating Greeks can be written as

$$\frac{\partial V}{\partial \theta} = \frac{\partial \mathbb{E}(P)}{\partial \theta} \approx \frac{\partial \mathbb{E}(P_L)}{\partial \theta} = \frac{\partial \mathbb{E}(P_0)}{\partial \theta} + \sum_{\ell=1}^L \frac{\partial \mathbb{E}(P_\ell^f - P_{\ell-1}^c)}{\partial \theta}. \quad (25)$$

Therefore extending Monte Carlo Greeks to MLMC Greeks is straightforward. However, the challenge is to keep the MLMC variance small. This can be achieved by appropriate smoothing of the payoff function. The techniques that were presented in section 3.2 are also very useful here.

4.3 European call

As an example we consider an European call $P = (x(T) - B)^+$ with $x(t)$ being a geometric Brownian motion with Milstein scheme approximation given by

$$X_{n+1}^\ell = X_n^\ell + r X_n^\ell \Delta t_\ell + \sigma X_n^\ell \Delta w_{n+1}^\ell + \frac{\sigma^2}{2} ((\Delta w_{n+1}^\ell)^2 - \Delta t_\ell). \quad (26)$$

We illustrate the techniques by computing delta (δ) and vega (ν), the sensitivities to the asset's initial value $x(0)$ and to its volatility σ .

Since the payoff is Lipschitz, we can use pathwise sensitivities. We observe that

$$\frac{\partial}{\partial x}(x - B)^+ = \begin{cases} 0, & \text{for } x < B \\ 1, & \text{for } x > B \end{cases}$$

This derivative fails to exist when $x = B$, but since this event has probability 0, we may write

$$\frac{\partial}{\partial x}(x - K)^+ = \mathbf{1}_{\{X > B\}}.$$

Therefore we are essentially dealing with a digital option.

4.4 Conditional Monte Carlo for Pathwise Sensitivity

Using conditional expectation the payoff can be smooth as we did it in Section 3.2. European calls can be treated in the exactly the same way as Digital option in Section 3.2, that is instead of simulating the whole path, we stop at the penultimate step and then on the last step we consider the full distribution of $(X_{2^\ell}^\ell \mid w_0^l, \dots, w_{2^\ell-1}^l)$.

For digital options this approach leads to (19) and (20). For the call options we can do analogous calculations. In [8] numerical results for this approach obtained, with scalar Milstein scheme used to obtain the penultimate step. They results are presented in Table 3. For lookback options conditional expectations leads to (13) and (15) and for barriers to (16) and (17). Burgos et al [8], applied pathwise sensitivity to these smoothed payoffs, with scalar Milstein scheme used to obtain the penultimate step, and obtained numerical results that we present in Table 4.

	Call		Digital	
Estimator	β	MLMC Complexity	β	MLMC Complexity
Value	≈ 2.0	$\mathcal{O}(\epsilon^{-2})$	≈ 1.4	$\mathcal{O}(\epsilon^{-2})$
Delta	≈ 1.5	$\mathcal{O}(\epsilon^{-2})$	≈ 0.5	$\mathcal{O}(\epsilon^{-2.5})$
Vega	≈ 2	$\mathcal{O}(\epsilon^{-2})$	≈ 0.6	$\mathcal{O}(\epsilon^{-2.4})$

Table 3: Orders of convergence for V_ℓ as observed numerically and corresponding MLMC complexity.

	Lookback		Barrier	
Estimator	β	MLMC Complexity	β	MLMC Complexity
Value	≈ 1.9	$\mathcal{O}(\epsilon^{-2})$	≈ 1.6	$\mathcal{O}(\epsilon^{-2})$
Delta	≈ 1.9	$\mathcal{O}(\epsilon^{-2})$	≈ 0.6	$\mathcal{O}(\epsilon^{-2.4})$
Vega	≈ 1.3	$\mathcal{O}(\epsilon^{-2})$	≈ 0.6	$\mathcal{O}(\epsilon^{-2.4})$

Table 4: Orders of convergence for V_ℓ as observed numerically and corresponding MLMC complexity.

4.5 Split pathwise sensitivities

There are two difficulties in using conditional expectation to smooth payoffs in practice in financial applications. This first is that conditional expectation will often become a multi-dimensional integral without an obvious closed-form value, and the second is that it requires a change to the often complex software framework used to specify payoffs. As a remedy for these problems the splitting technique to approximate $\mathbb{E} [P(X_{2^\ell}^\ell) \mid X_{2^\ell-1}^\ell]$ and

$\mathbb{E} \left[P(X_{2^{\ell-1}}^{\ell-1}) \mid X_{2^{\ell-1}-1}^{\ell-1}, \Delta w_{2^{\ell-2}}^{\ell} \right]$, is used. We get numerical estimates of these values by “splitting” every simulated path on the final timestep. At the fine level: for every simulated path, a set of s final increments $\{\Delta w_{2^\ell}^{\ell,i}\}_{i \in [1,s]}$ is simulated, which can be averaged to get

$$\mathbb{E} \left[P(X_{2^\ell}^\ell) \mid X_{2^\ell-1}^\ell \right] \approx \frac{1}{s} \sum_{i=1}^s P(X_{2^\ell-1}^\ell, \Delta w_{2^\ell}^{\ell,i}) \quad (27)$$

At the coarse level, similar to the case of digital options, the fine increment of the Brownian motion over the first half of the coarse timestep is used,

$$\mathbb{E} \left[P(X_{2^{\ell-1}}^{\ell-1}) \mid X_{2^{\ell-1}-1}^{\ell-1}, \Delta w_{2^{\ell-2}}^\ell, \Delta w_{2^{\ell-1}}^{\ell-1,i} \right] \approx \frac{1}{s} \sum_{i=1}^s P(X_{2^{\ell-1}-1}^{\ell-1}, \Delta w_{2^{\ell-2}}^\ell, \Delta w_{2^{\ell-1}}^{\ell-1,i}) \quad (28)$$

This approach was tested in [8], with scalar the Milstein scheme used to obtain the penultimate step, and is presented in Table 5. As expected the values of β tend to the rates offered by conditional expectations as s increases and the approximation gets more precise.

Estimator	s	β	MLMC Complexity
Value	10	≈ 2.0	$O(\epsilon^{-2})$
	500	≈ 2.0	$O(\epsilon^{-2})$
Delta	10	≈ 1.0	$O(\epsilon^{-2}(\log \epsilon)^2)$
	500	≈ 1.5	$O(\epsilon^{-2})$
Vega	10	≈ 1.6	$O(\epsilon^{-2})$
	500	≈ 2.0	$O(\epsilon^{-2})$

Table 5: Orders of convergence for V_ℓ as observed numerically and the corresponding MLMC complexity.

4.6 Optimal number of samples

The use of multiple samples to estimate the value of the conditional expectations is an example of the splitting technique [2]. If w and z are independent random variables, then for any function $P(w, z)$ the estimator

$$Y_{M,S} = M^{-1} \sum_{m=1}^M \left(S^{-1} \sum_{i=1}^S P(w^m, z^{(m,i)}) \right)$$

with independent samples w^m and $z^{m,i}$ is an unbiased estimator for

$$\mathbb{E}_{w,z} [P(w, z)] \equiv \mathbb{E}_w [\mathbb{E}_z [P(w, z) \mid w]],$$

and its variance is

$$\mathbb{V}[Y_{M,S}] = M^{-1} \mathbb{V}_w [\mathbb{E}_z [P(w, z) \mid w]] + (MS)^{-1} \mathbb{E}_w [\mathbb{V}_z [P(w, z) \mid w]].$$

The cost of computing $Y_{M,S}$ with variance $v_1 M^{-1} + v_2 (MS)^{-1}$ is proportional to

$$c_1 M + c_2 MS,$$

with c_1 corresponding to the path calculation and c_2 corresponding to the payoff evaluation. For a fixed computational cost, the variance can be minimised by minimising the product

$$(v_1 + v_2 s^{-1}) (c_1 + c_2 s) = v_1 c_2 s + v_1 c_1 + v_2 c_2 + v_2 c_1 s^{-1},$$

which gives the optimum value $s_{\text{opt}} = \sqrt{v_2 c_1 / v_1 c_2}$.

c_1 is $\mathcal{O}(\Delta t_\ell^{-1})$ since the cost is proportional to the number of timesteps, and c_2 is $\mathcal{O}(1)$, independent of Δt_ℓ . If the payoff is Lipschitz, then v_1 and v_2 are both $\mathcal{O}(1)$ and $S_{\text{opt}} = \mathcal{O}(\Delta t_\ell^{-1/2})$.

4.7 Vibrato Monte Carlo

The idea of vibrato Monte Carlo is to combine pathwise sensitivity and Likelihood Ratio Method. Adopting the conditional expectation approach, each path simulation for a particular set of Brownian motion increments $w^\ell \equiv (\Delta w_1^\ell, \Delta w_2^\ell, \dots, \Delta w_{2^\ell-1}^\ell)$ (excluding the increment for the final timestep) computes a conditional Gaussian probability distribution $p_X(X_{2^\ell}^\ell | w^\ell)$. For a scalar SDE, if μ_{w^ℓ} and σ_{w^ℓ} are the mean and standard deviation for given w^ℓ , then

$$X_{2^\ell}^\ell(w^\ell, Z) = \mu_{w^\ell} + \sigma_{w^\ell} Z,$$

where Z is a unit Normal random variable. The expected payoff can then be expressed as

$$V_\ell = \mathbb{E} \left[\mathbb{E} [P(X_{2^\ell}^\ell) | w^\ell] \right] = \int \left\{ \int P(x) p_{X_{2^\ell}^\ell}(x | w^\ell) dx \right\} p_{w^\ell}(y) dy.$$

The outer expectation is an average over the discrete Brownian motion increments, while the inner conditional expectation is averaging over Z .

To compute the sensitivity to the input parameter θ , the first step is to apply the pathwise sensitivity approach for fixed w^l to obtain $\partial \mu_{w^l} / \partial \theta, \partial \sigma_{w^l} / \partial \theta$. We then apply LRM to the inner conditional expectation to get

$$\frac{\partial V_\ell}{\partial \theta} = \mathbb{E} \left[\frac{\partial}{\partial \theta} \mathbb{E} [P(X_{2^\ell}^\ell) | w^\ell] \right] = \mathbb{E} \left[\mathbb{E}_Z \left[P(X_{2^\ell}^\ell) \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \theta} | w^\ell \right] \right],$$

where

$$\frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \theta} = \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \mu_{w^\ell}} \frac{\partial \mu_{w^\ell}}{\partial \theta} + \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \sigma_{w^\ell}} \frac{\partial \sigma_{w^\ell}}{\partial \theta}.$$

This leads to the estimator

$$\begin{aligned} \frac{\partial V_\ell}{\partial \theta} \approx \frac{1}{N_\ell} \sum_{m=1}^{N_\ell} & \left(\frac{\partial \mu_{w^\ell, m}}{\partial \theta} \mathbb{E} \left[P(X_{2^\ell}^\ell) \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \mu_{w^\ell}} | w^{\ell, m} \right] \right. \\ & \left. + \frac{\partial \sigma_{\hat{w}^{\ell, m}}}{\partial \theta} \mathbb{E} \left[P(X_{2^\ell}^\ell) \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \sigma_{w^\ell}} | w^{\ell, m} \right] \right) \end{aligned} \quad (29)$$

We compute $\frac{\partial \mu_{w^\ell, m}}{\partial \theta}$ and $\frac{\partial \sigma_{w^\ell, m}}{\partial \theta}$ with pathwise sensitivities. With $X_{2^i}^{\ell, m, i} = X_{2^\ell}^\ell(w^{\ell, m}, Z^i)$, we substitute the following estimators into (29)

$$\begin{cases} \mathbb{E} \left[P(X_{2^\ell}^\ell) \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \mu_{w^\ell}} | w^{\ell, m} \right] \approx \frac{1}{s} \sum_{i=1}^s \left(P(X_{2^\ell}^{\ell, m, i}) \frac{X_{2^\ell}^{2^\ell, m, i} - \mu_{w^\ell, m}}{\sigma_{w^\ell, m}^2} \right) \\ \mathbb{E} \left[P(X_{2^\ell}^\ell) \frac{\partial(\log p_{X_{2^\ell}^\ell})}{\partial \sigma_{w^\ell}} | \hat{w}^{\ell, m} \right] \approx \frac{1}{s} \sum_{i=1}^s P(X_{2^\ell}^{\ell, m, i}) \left(-\frac{1}{\sigma_{w^\ell, m}} + \frac{(X_{2^\ell}^{\ell, m, i} - \mu_{w^\ell, m})^2}{\sigma_{w^\ell, m}^3} \right) \end{cases}$$

In a multilevel setting, at the fine level we can use (29) directly. At the coarse level, as for digital options in section 3.5, the fine Brownian increments over the first half of the coarse timestep are re-used to derive (29).

The numerical experiments for the call option with $s = 10$ was obtained [8], with scalar Milstein scheme used to obtain the penultimate step.

Estimator	β	MLMC Complexity
Value	≈ 2.0	$O(\epsilon^{-2})$
Delta	≈ 1.5	$O(\epsilon^{-2})$
Vega	≈ 2.0	$O(\epsilon^{-2})$

Although the discussion so far has considered an option based on the value of a single underlying value at the terminal time T , it can be shown that the idea extends very naturally to multidimensional cases, producing a conditional multivariate Gaussian distribution, and also to financial payoffs which are dependent on values at intermediate times.

5 MLMC for Jump-diffusion processes

Giles and Xia in [47] investigated the extension of the MLMC method to jump-diffusion SDEs. We consider models with finite rate activity using a jump-adapted discretisation in which the jump times are computed and added to the standard uniform discretisation times. If the Poisson jump rate is constant, the jump times are the same on both paths and the multilevel extension is relatively straightforward, but the implementation is more complex in the case of state-dependent jump rates for which the jump times naturally differ.

Merton[36] proposed a jump-diffusion process, in which the asset price follows a jump-diffusion SDE:

$$dx(t) = f(x(t-)) \Delta t + g(x(t-)) \Delta w(t) + c(x(t-)) \Delta J(t), \quad 0 \leq t \leq T, \quad (30)$$

where the jump term $J(t)$ is a compound Poisson process $\sum_{i=1}^{N(t)} (Y_i - 1)$, the jump magnitude Y_i has a prescribed distribution, and $N(t)$ is a Poisson process with intensity λ , independent of the Brownian motion. Due to the existence of jumps, the process is a càdlàg process, i.e. having right continuity with left limits. We note that $x(t-)$ denotes the left limit of the process while $x(t) = \lim_{s \rightarrow t+} x(s)$. In [36], Merton also assumed that $\log Y_i$ has a normal distribution.

5.1 A Jump-adapted Milstein discretisation

To simulate finite activity jump-diffusion processes, Giles and Xia [47] used the jump-adapted approximation from Platen and Bruti-Liberat[40]. For each path simulation, the set of jump times $\mathbb{J} = \{\tau_1, \tau_2, \dots, \tau_m\}$ within the time interval $[0, T]$ is added to a uniform partition $\mathcal{P}_{\Delta t_l} := \{n\Delta t_l : n = 0, 1, 2, \dots, 2^l\}$. A combined set of discretisation times is then given by $\mathbb{T} = \{0 = t_0 < t_1 < t_2 < \dots < t_M = T\}$ and we define the length of the timestep as $\Delta t_l^n = t_{n+1} - t_n$. Clearly, $\Delta t_l^n \leq \Delta t_l$.

Within each timestep the scalar Milstein discretisation is used to approximate the SDE (30), and then the jump is simulated when the simulation time is equal to one of the jump times. This gives the following numerical method:

$$\begin{aligned} X_{n+1}^{\ell,-} &= X_n^\ell + f(X_n^\ell) \Delta t_\ell^n + g(X_n^\ell) \Delta w_{n+1}^\ell + \frac{1}{2} g'(X_n^\ell) g(X_n^\ell) (\Delta (w_n^\ell)^2 - \Delta t_\ell^n), \\ X_{n+1}^\ell &= \begin{cases} X_{n+1}^{\ell,-} + c(X_{n+1}^{\ell,-})(Y_i - 1), & \text{when } t_{n+1} = \tau_i; \\ X_{n+1}^{\ell,-}, & \text{otherwise,} \end{cases} \end{aligned} \quad (31)$$

where $X_n^{\ell,-} = X_{t_n-}^\ell$ is the left limit of the approximated path, Δw_n^ℓ is the Brownian increment and Y_i is the jump magnitude at τ_i .

5.1.1 Multilevel Monte Carlo for constant jump rate

In the case of the jump-adapted discretisation the telescopic sum (1) is written down with respect to Δt_ℓ rather than to Δt_ℓ^n . Therefore, we have to define the computational complexity as the expected computational cost since different paths may have different numbers of jumps. However, the expected number of jumps is finite and therefore the cost bound in assumption *iv*) will still remain valid for an appropriate choice of the constant c_3 .

The MLMC approach for a constant jump rate is straightforward. The jump times τ_j , which are the same for the coarse and fine paths, are simulated by setting $\tau_j - \tau_{j-1} \sim \exp(\lambda)$.

Pricing European call and Asian options in this setting is straightforward. For lookback, barrier and digital options we need to consider Brownian bridge interpolations as we did in Section 3.2. However, due to presence of jumps some small modifications are required. To improve convergence we will be looking at Brownian bridges between time-steps coming from jump-adapted discretization. In order to obtain an interpolated value $\tilde{X}_{n+\frac{1}{2}}^{2^{\ell-1}}$ for the coarse time-step a Brownian Bridge interpolation over interval $[k_n, \hat{k}_n]$ is considered, where

$$\begin{aligned} k_n &= \max \{n\Delta t_{\ell-1}^n, \max \{\tau \in \mathbb{J} : \tau < (n + \tfrac{1}{2})\Delta t_{\ell-1}^n\}\} \\ \hat{k}_n &= \min \{(n+1)\Delta t_{\ell-1}^n, \min \{\tau \in \mathbb{J} : \tau > (n + \tfrac{1}{2})\Delta t_{\ell-1}^n\}\}. \end{aligned} \quad (32)$$

Hence

$$\begin{aligned} \tilde{X}_{n+\frac{1}{2}}^{\ell-1} &= X_{k_n}^{\ell-1} + \lambda_{\ell-1} (X_{\hat{k}_n}^{\ell-1} - X_{k_n}^{\ell-1}) \\ &\quad + g(X_{k_n}^{\ell-1}) \left(w^\ell((n + \tfrac{1}{2})\Delta t_{\ell-1}) - w^\ell(k_n) - \lambda_{\ell-1} (w^\ell(\hat{k}_n) - w^\ell(k_n)) \right) \end{aligned}$$

where $\lambda_{\ell-1} \equiv ((n + \frac{1}{2})\Delta t_{\ell-1}^n - k_n)/(\hat{k}_n - k_n)$.

In the same way as in Section 3.2, the minima over time-adapted discretization can be derived. For the fine time-step we have

$$X_{n,min}^\ell = \tfrac{1}{2} \left(X_n^\ell + X_{n+\frac{1}{2}}^{\ell,-} - \sqrt{\left(X_{n+\frac{1}{2}}^{\ell,-} - X_n^\ell \right)^2 - 2g(X_n^\ell)^2 \Delta t_\ell^n \log U_n^\ell} \right).$$

Notice the use of the left limits $X^{\ell,-}$. Following discussion in the previous sections, the minima for the coarse time-step can be derived using interpolated value $\tilde{X}_{n+\frac{1}{2}}^{\ell-1}$. Deriving the payoffs for lookback and barrier option is now straightforward.

For digital options, due to jump-adapted time grid, in order to find conditional expectations, we need to look at relations between the last jump time and the last timestep before expiry. In fact, there are three cases:

1. The last jump time τ happens before penultimate fixed-time timestep, i.e. $\tau < (2^{l-1} - 2)\Delta t_l$.
2. The last jump time is within the last fixed-time timestep, i.e. $\tau > (2^{l-1} - 1)\Delta t_l$;
3. The last jump time is within the penultimate fixed-time timestep, i.e. $(2^{l-1} - 1)\Delta t_l > \tau > (2^{l-1} - 2)\Delta t_l$.

With this in mind we can easily write down the payoffs for the coarse and fine approximations as we presented in Section 3.5.

5.1.2 MLMC for Path-dependent rates

In the case of a path-dependent jump rate $\lambda(x(t))$, the implementation of the multilevel method becomes more difficult because the coarse and fine path approximations may have jumps at different times. These differences could lead to a large difference between the coarse and fine path payoffs, and hence greatly increase the variance of the multilevel correction. To avoid this, Giles and Xia [47] modified the simulation approach of Glasserman and Merener [26] which uses “thinning” to treat the case in which $\lambda(x(t), t)$ is bounded. Let us recall the thinning property of Poisson processes. Let $(N_t)_{t \geq 0}$ be a Poisson process with intensity λ and define a new process Z_t by “thinning” N_t : take all the jump times $(\tau_n, n \geq 1)$ corresponding to N , keep then with probability $0 < p < 1$ or delete then with probability $1 - p$, independently from each other. Now order the jump times that have not been deleted: $(\tau'_n, n \geq 1)$, and define

$$Z_t = \sum_{n \geq 1} \mathbf{1}_{t \geq \tau'_n}.$$

Then the process Z is Poisson process with intensity $p\lambda$.

In our setting, first a Poisson process with a constant rate λ_{sup} (which is an upper bound of the state-dependent rate) is constructed. This gives a set of candidate jump times, and these are then selected as true jump times with probability $\lambda(x(t), t)/\lambda_{\text{sup}}$. The following jump-adapted thinning Milstein scheme is obtained

1. Generate the jump-adapted time grid for a Poisson process with constant rate λ_{sup} ;
2. Simulate each timestep using the Milstein discretisation;
3. When the endpoint t_{n+1} is a candidate jump time, generate a uniform random number $U \sim [0, 1]$, and if $U < p_{t_{n+1}} = \frac{\lambda(x(t_{n+1}-), t_{n+1})}{\lambda_{\text{sup}}}$, then accept t_{n+1} as a real jump time and simulate the jump.

In the multilevel implementation, the straightforward application of the above algorithm will result in different acceptance probabilities for fine and coarse level. There may be some samples in which a jump candidate is accepted for the fine path, but not for the coarse path, or vice versa. Because of the first order strong convergence, the difference in acceptance probabilities will be $\mathcal{O}(\Delta t_\ell)$, and hence there is an $\mathcal{O}(\Delta t_\ell)$ probability of coarse and fine paths differing in accepting candidate jumps. Such differences will give an $\mathcal{O}(1)$ difference in the payoff value, and hence the multilevel variance will be $\mathcal{O}(h)$. A more detailed analysis of this is given in [46].

To improve the variance convergence rate, a change of measure is used so that the acceptance probability is the same for both fine and coarse paths. This is achieved by taking the expectation with respect to a new measure Q :

$$\mathbb{E}_P[P_\ell^f - P_{\ell-1}^c] = \mathbb{E}_Q[P_\ell^f \prod_{\tau} R_\tau^f - P_{\ell-1}^c \prod_{\tau} R_\tau^c]$$

where τ are the jump times. The acceptance probability for a candidate jump under the measure Q is defined to be $\frac{1}{2}$ for both coarse and fine paths, instead of $p_\tau = \lambda(X(\tau-), \tau) / \lambda_{\text{sup}}$. The corresponding Radon-Nikodym derivatives are

$$R_\tau^f = \begin{cases} 2p_\tau^f, & \text{if } U < \frac{1}{2}; \\ 2(1 - p_\tau^f), & \text{if } U \geq \frac{1}{2}, \end{cases} \quad R_\tau^c = \begin{cases} 2p_\tau^c, & \text{if } U < \frac{1}{2}; \\ 2(1 - p_\tau^c), & \text{if } U \geq \frac{1}{2}, \end{cases}$$

Since $\mathbb{V}[R_\tau^f - R_\tau^c] = \mathcal{O}(\Delta t^2)$ and $\mathbb{V}[\hat{P}_\ell - \hat{P}_{\ell-1}] = \mathcal{O}(\Delta t^2)$, this results in the multilevel correction variance $\mathbb{V}_Q[\hat{P}_\ell \prod_\tau R_\tau^f - \hat{P}_{\ell-1} \prod_\tau R_\tau^c]$ being $\mathcal{O}(\Delta t^2)$.

If the analytic formulation is expressed using the same thinning and change of measure, the weak error can be decomposed into two terms as follows:

$$\mathbb{E}_Q \left[\hat{P}_\ell \prod_\tau R_\tau^f - P \prod_\tau R_\tau \right] = \mathbb{E}_Q \left[(\hat{P}_\ell - P) \prod_\tau R_\tau^f \right] + \mathbb{E}_Q \left[P \left(\prod_\tau R_\tau^f - \prod_\tau R_\tau \right) \right].$$

Using Hölder's inequality, the bound $\max(R_\tau, R_\tau^f) \leq 2$ and standard results for a Poisson process, the first term can be bounded using weak convergence results for the constant rate process, and the second term can be bounded using the corresponding strong convergence results [46]. This guarantees that the multilevel procedure does converge to the correct value.

5.2 Lévy processes

Dereich and Heidenreich [13] analysed approximation methods for both finite and infinite activity Lévy driven SDEs with globally Lipschitz payoffs. They have derived upper bounds for MLMC variance for the class of path dependent payoffs that are Lipschitz continuous with respect to supremum norm. One of their main findings is that the rate of MLMC variance converges is closely related to Blumenthal-Gettoor index of the driving Lévy process that measures the frequency of small jumps. In [13] authors considered SDEs driven by the Lévy process

$$s(t) = \Sigma w(t) + L(t) + bt,$$

where Σ is the diffusion coefficient, $L(t)$ is a compensated jump process and b is a drift coefficient. The simplest treatment is to neglect all the jumps with size smaller than h . To construct MLMC they took h^ℓ , that is at level ℓ they neglected jumps smaller than h^ℓ . Then similarly as in the previous section, a uniform time discretization Δt_ℓ augmented with jump times is used. Let us denote by $\Delta L(t) = L(t) - L(t)^-$, the jump-discontinuity at time t . The crucial observation is that for $h' > h > 0$ the jumps of the process $L^{h'}$ can be obtained from those of L^h by

$$\Delta L(t)^{h'} = \Delta L_t^h \mathbf{1}_{\{|\Delta L(t)^h| > h'\}},$$

this gives the necessary coupling to obtain a good MLMC variance. We define a decreasing and invertible function $g : (0, \infty) \rightarrow (0, \infty)$ such that

$$\int \frac{|x|^2}{h^2} \wedge 1 \nu(dx) \leq g(h) \quad \text{for all } h > 0,$$

where ν is a Lévy measure, and for $\ell \in \mathbb{N}$ we define

$$\Delta t_\ell = 2^{-\ell} \quad \text{and} \quad h^\ell = g^{-1}(2^\ell).$$

With this choice of Δt_ℓ and h^ℓ , authors in [13] analysed the standard Euler-Maruyama scheme for Lévy driven SDEs. This approach gives good results for a Blumenthal-Gettoor index smaller than one. For a Blumenthal-Gettoor index bigger than one, Gaussian approximation of small jumps gives better results [12].

6 Multi-dimensional Milstein scheme

In the previous sections it was shown that by combining a numerical approximation with the strong order of convergence $\mathcal{O}(\Delta t_\ell)$ with MLMC results in reduction of the computational complexity to estimate expected values of functionals of SDE solutions with a root-mean-square error of ϵ from $\mathcal{O}(\epsilon^{-3})$ to $\mathcal{O}(\epsilon^{-2})$. However, in general, to obtain a rate of strong convergence higher than $O(\Delta t^{1/2})$ requires simulation, or approximation, of Lévy areas. Giles and Szpruch in [22] through the construction of a suitable antithetic multi-level correction estimator, showed that we can avoid the simulation of Lévy areas and still achieve an $O(\Delta t^2)$ variance for smooth payoffs, and almost an $O(\Delta t^{3/2})$ variance for piecewise smooth payoffs, even though there is only $O(\Delta t^{1/2})$ strong convergence.

In the previous sections we have shown that it can be better to use different estimators for the finer and coarser of the two levels being considered, P_ℓ^f when level ℓ is the finer level, and P_ℓ^c when level ℓ is the coarser level. In this case, we required that

$$\mathbb{E}[P_\ell^f] = \mathbb{E}[P_\ell^c] \quad \text{for } \ell = 1, \dots, L, \tag{33}$$

so that

$$\mathbb{E}[P_L^f] = \mathbb{E}[P_0^f] + \sum_{\ell=1}^L \mathbb{E}[P_\ell^f - P_{\ell-1}^c],$$

still holds. For lookback, barrier and digital options we showed that we can obtain a better MLMC variance by suitable modifying the estimator on the coarse levels. By further exploiting the flexibility of MLMC, Giles and Szpruch [22] modified the estimator on the fine levels in order to avoid simulation of the Lévy areas.

6.1 Antithetic MLMC estimator

Based on the well-known method of antithetic variates (see for example [24]), the idea for the antithetic estimator is to exploit the flexibility of the more general MLMC estimator by defining $P_{\ell-1}^c$ to be the usual payoff $P(X^c)$ coming from a level $\ell-1$ coarse simulation X^c , and defining P_ℓ^f to be the average of the payoffs $P(X^f), P(X^a)$ coming from an antithetic pair of level ℓ simulations, X^f and X^a .

X^f will be defined in a way which corresponds naturally to the construction of X^c . Its antithetic “twin” X^a will be defined so that it has exactly the same distribution as X^f , conditional on X^c , which ensures that $\mathbb{E}[P(X^f)] = \mathbb{E}[P(X^a)]$ and hence (3) is satisfied, but at the same time

$$(X^f - X^c) \approx -(X^a - X^c)$$

and therefore

$$(P(X^f) - P(X^c)) \approx -(P(X^a) - P(X^c)),$$

so that $\frac{1}{2}(P(X^f) + P(X^a)) \approx P(X^c)$. This leads to $\frac{1}{2}(P(X^f) + P(X^a)) - P(X^c)$ having a much smaller variance than the standard estimator $P(X^f) - P(X^c)$.

We now present a lemma which gives an upper bound on the convergence of the variance of $\frac{1}{2}(P(X^f) + P(X^a)) - P(X^c)$.

Lemma 6.1. *If $P \in C^2(\mathbb{R}^d, \mathbb{R})$ and there exist constants L_1, L_2 such that for all $x \in \mathbb{R}^d$*

$$\left\| \frac{\partial P}{\partial x} \right\| \leq L_1, \quad \left\| \frac{\partial^2 P}{\partial x^2} \right\| \leq L_2.$$

then for $p \geq 2$,

$$\begin{aligned} & \mathbb{E} \left[\left(\frac{1}{2}(P(X^f) + P(X^a)) - P(X^c) \right)^p \right] \\ & \leq 2^{p-1} L_1^p \mathbb{E} \left[\left\| \frac{1}{2}(X^f + X^a) - X^c \right\|^p \right] + 2^{-(p+1)} L_2^p \mathbb{E} \left[\|X^f - X^a\|^{2p} \right]. \end{aligned}$$

In the multidimensional SDE applications considered in finance, the Milstein approximation with the Lévy areas set to zero, combined with the antithetic construction, leads to $X^f - X^a = O(\Delta t^{1/2})$ but $\bar{X}^f - X^c = O(\Delta t)$. Hence, the variance $\mathbb{V}[\frac{1}{2}(P_\ell^f + P_\ell^a) - P_{\ell-1}^c]$ is $O(\Delta t^2)$, which is the order obtained for scalar SDEs using the Milstein discretisation with its first order strong convergence.

6.2 Clark-Cameron Example

The paper of Clark and Cameron [9] addresses the question of how accurately one can approximate the solution of an SDE driven by an underlying multi-dimensional Brownian motion, using only uniformly-spaced discrete Brownian

increments. Their model problem is

$$\begin{aligned} dx_1(t) &= dw_1(t) \\ dx_2(t) &= x_1(t) dw_2(t), \end{aligned} \quad (34)$$

with $x(0) = y(0) = 0$, and zero correlation between the two Brownian motions $w_1(t)$ and $w_2(t)$. These equations can be integrated exactly over a time interval $[t_n, t_{n+1}]$, where $t_n = n \Delta t$, to give

$$\begin{aligned} x_1(t_{n+1}) &= x_1(t_n) + \Delta w_{1,n} \\ x_2(t_{n+1}) &= x_2(t_n) + x_1(t_n) \Delta w_{2,n} + \frac{1}{2} \Delta w_{1,n} \Delta w_{2,n} + \frac{1}{2} A_{12,n} \end{aligned} \quad (35)$$

where $\Delta w_{i,n} \equiv w_i(t_{n+1}) - w_i(t_n)$, and $A_{12,n}$ is the Lévy area defined as

$$A_{12,n} = \int_{t_n}^{t_{n+1}} (w_1(t) - w_1(t_n)) dw_2(t) - \int_{t_n}^{t_{n+1}} (w_2(t) - w_2(t_n)) dw_1(t).$$

This corresponds exactly to the Milstein discretisation presented in (7), so for this simple model problem the Milstein discretisation is exact.

The point of Clark and Cameron's paper is that for *any* numerical approximation $X(T)$ based solely on the set of discrete Brownian increments Δw ,

$$\mathbb{E}[(x_2(T) - X_2(T))^2] \geq \frac{1}{4} T \Delta t.$$

Since in this section we use superscript f, a, c for fine X^f , antithetic X^a and coarse X^c approximations, respectively, we drop the superscript ℓ for the clarity of notation.

We define a coarse path approximation X^c with timestep Δt by neglecting the Lévy area terms to give

$$\begin{aligned} X_{1,n+1}^c &= X_{1,n}^c + \Delta w_{1,n+1}^{\ell-1} \\ X_{2,n+1}^c &= X_{2,n}^c + X_{1,n}^c \Delta w_{2,n+1}^{\ell-1} + \frac{1}{2} \Delta w_{1,n+1}^{\ell-1} \Delta w_{2,n+1}^{\ell-1} \end{aligned} \quad (36)$$

This is equivalent to replacing the true Brownian path by a piecewise linear approximation as illustrated in Figure 1.

Similarly, we define the corresponding two half-timesteps of the first fine path approximation X^f by

$$\begin{aligned} X_{1,n+\frac{1}{2}}^f &= X_{1,n}^f + \Delta w_{1,n+\frac{1}{2}}^\ell \\ X_{2,n+\frac{1}{2}}^f &= X_{2,n}^f + X_{1,n}^f \Delta w_{2,n+\frac{1}{2}}^\ell + \frac{1}{2} \Delta w_{1,n+\frac{1}{2}}^\ell \Delta w_{2,n+\frac{1}{2}}^\ell \\ X_{1,n+1}^f &= X_{1,n+1}^f + \Delta w_{1,n+1}^\ell \\ X_{2,n+1}^f &= X_{2,n+\frac{1}{2}}^f + X_{1,n+\frac{1}{2}}^f \Delta w_{2,n+1}^\ell + \frac{1}{2} \Delta w_{1,n+1}^\ell \Delta w_{2,n+1}^\ell \end{aligned}$$

where $\Delta w_{n+1}^{\ell-1} = \Delta w_{n+\frac{1}{2}}^\ell + \Delta w_{n+1}^\ell$. Using this relation, the equations for the two fine timesteps can be combined to give an equation for the increment over

the coarse timestep,

$$\begin{aligned}
X_{1,n+1}^f &= X_{1,n}^f + \Delta w_{1,n+1}^{\ell-1} \\
X_{2,n+1}^f &= X_{2,n}^f + X_{1,n}^f \Delta w_{2,n+1}^{\ell-1} + \frac{1}{2} \Delta w_{1,n+1}^{\ell-1} \Delta w_{2,n+1}^{\ell-1} \\
&\quad + \frac{1}{2} \left(\Delta w_{1,n+\frac{1}{2}}^\ell \Delta w_{2,n+1}^\ell - \Delta w_{2,n+\frac{1}{2}}^\ell \Delta w_{1,n+1}^\ell \right).
\end{aligned} \tag{37}$$

The antithetic approximation X_n^a is defined by exactly the same discretisation except that the Brownian increments δw_n and $\delta w_{n+\frac{1}{2}}$ are swapped, as illustrated in Figure 1. This gives

$$\begin{aligned}
X_{1,n+\frac{1}{2}}^a &= X_{1,n}^a + \Delta w_{1,n+1}^\ell, \\
X_{2,n+\frac{1}{2}}^a &= X_{2,n}^a + X_{1,n}^a \Delta w_{2,n+1}^\ell + \frac{1}{2} \Delta w_{1,n+1}^\ell \Delta w_{2,n+1}^\ell, \\
X_{1,n+1}^a &= X_{1,n+\frac{1}{2}}^a + \Delta w_{1,n+\frac{1}{2}}^\ell, \\
X_{2,n+1}^a &= X_{2,n+\frac{1}{2}}^a + X_{1,n+\frac{1}{2}}^a \Delta w_{2,n+\frac{1}{2}}^\ell + \frac{1}{2} \Delta w_{1,n+\frac{1}{2}}^\ell \Delta w_{2,n+\frac{1}{2}}^\ell,
\end{aligned}$$

and hence

$$\begin{aligned}
X_{1,n+1}^a &= X_{1,n}^a + \Delta w_{1,n+1}^{\ell-1}, \\
X_{2,n+1}^a &= X_{2,n}^a + X_{1,n}^a \Delta w_{2,n+1}^{\ell-1} + \frac{1}{2} \Delta w_{1,n+1}^{\ell-1} \Delta w_{2,n+1}^{\ell-1} \\
&\quad - \frac{1}{2} \left(\Delta w_{1,n+\frac{1}{2}}^\ell \Delta w_{2,n+1}^\ell - \Delta w_{2,n+\frac{1}{2}}^\ell \Delta w_{1,n+1}^\ell \right).
\end{aligned} \tag{38}$$

Swapping $\Delta w_{n+\frac{1}{2}}^\ell$ and Δw_{n+1}^ℓ does not change the distribution of the driving Brownian increments, and hence X^a has exactly the same distribution as X^f . Note also the change in sign in the last term in (37) compared to the corresponding term in (38). This is important because these two terms cancel when the two equations are averaged.

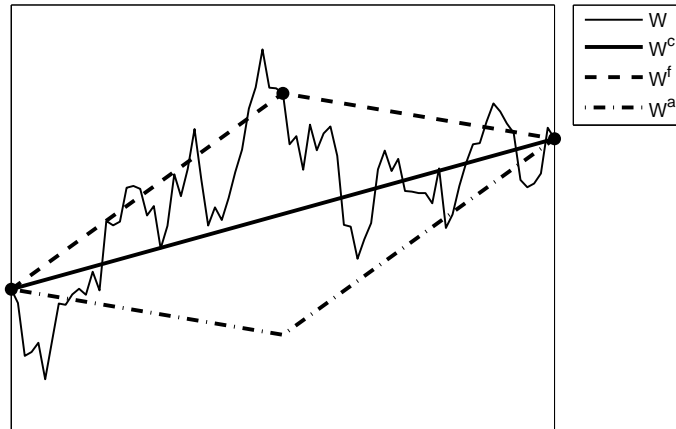


Figure 1: Brownian path and approximations over one coarse timestep

These last terms correspond to the Lévy areas for the fine path and the antithetic path, and the sign reversal is a particular instance of a more general result for time-reversed Brownian motion, [30]. If $(w_t, 0 \leq t \leq 1)$ denotes a Brownian motion on the time interval $[0, 1]$ then the time-reversed Brownian motion $(z_t, 0 \leq t \leq 1)$ defined by

$$z_t = w_1 - w_{1-t}, \quad (39)$$

has exactly the same distribution, and it can be shown that its Lévy area is equal in magnitude and opposite in sign to that of w_t .

Lemma 6.2. *If X_n^f , X_n^a and X_n^c are as defined above, then*

$$X_{1,n}^f = X_{1,n}^a = X_{1,n}^c, \quad \frac{1}{2} \left(X_{2,n}^f + X_{2,n}^a \right) = X_{2,n}^c, \quad \forall n \leq N$$

and

$$\mathbb{E} \left[\left(X_{2,N}^f - X_{2,N}^a \right)^4 \right] = \frac{3}{4} T (T + \Delta t) \Delta t^2.$$

In the next section we will see how this lemma generalizes to non-linear multidimensional SDEs (4).

6.3 Milstein discretisation - General theory

Using the coarse timestep Δt , the coarse path approximation X_n^c , is given by the Milstein approximation without the Lévy area term,

$$\begin{aligned} X_{i,n+1}^c &= X_{i,n}^c + f_i(X_n^c) \Delta t_{\ell-1} + \sum_{j=1}^m g_{ij}(X_n^c) \Delta w_{j,n+1}^{\ell-1} \\ &\quad + \sum_{j,k=1}^m h_{ijk}(X_n^c) \left(\Delta w_{j,n} \Delta w_{k,n+1}^{\ell-1} - \Omega_{jk} \Delta t_{\ell-1} \right). \end{aligned}$$

The first fine path approximation X_n^f (that corresponds to X_n^c) uses the corresponding discretisation with timestep $\Delta t/2$,

$$X_{i,n+\frac{1}{2}}^f = X_{i,n}^f + f_i(X_n^f) \Delta t_{\ell-1}/2 + \sum_{j=1}^m g_{ij}(X_n^f) \Delta w_{j,n+\frac{1}{2}}^\ell \quad (40)$$

$$+ \sum_{j,k=1}^m h_{ijk}(X_n^f) \left(\Delta w_{j,n+\frac{1}{2}}^\ell \Delta w_{k,n+\frac{1}{2}}^\ell - \Omega_{jk} \Delta t_{\ell-1}/2 \right),$$

$$X_{i,n+1}^f = X_{i,n+\frac{1}{2}}^f + f_i(X_{n+\frac{1}{2}}^f) \Delta t_{\ell-1}/2 + \sum_{j=1}^m g_{ij}(X_{n+\frac{1}{2}}^f) \Delta w_{j,n+1}^\ell \quad (41)$$

$$+ \sum_{j,k=1}^m h_{ijk}(X_{n+\frac{1}{2}}^f) \left(\Delta w_{j,n+1}^\ell \Delta w_{k,n+1}^\ell - \Omega_{jk} \Delta t_{\ell-1}/2 \right),$$

where $\Delta w_{n+1}^{\ell-1} = \Delta w_{n+\frac{1}{2}}^\ell + \Delta w_{n+1}^\ell$.

The antithetic approximation X_n^a is defined by exactly the same discretisation except that the Brownian increments $\Delta w_{n+\frac{1}{2}}^\ell$ and Δw_{n+1}^ℓ are swapped, so that

$$\begin{aligned} X_{i,n+\frac{1}{2}}^a &= X_{i,n}^a + f_i(X_n^a) \Delta t_{\ell-1}/2 + \sum_{j=1}^m g_{ij}(X_n^a) \delta w_{n+\frac{1}{2}} \\ &\quad + \sum_{j,k=1}^m h_{ijk}(X_n^a) (\Delta w_{j,n+\frac{1}{2}}^\ell \Delta w_{k,n+1}^\ell - \Omega_{jk} \Delta t_{\ell-1}/2), \\ X_{i,n+1}^a &= X_{i,n+\frac{1}{2}}^a + f_i(X_{n+\frac{1}{2}}^a) \Delta t_{\ell-1}/2 + \sum_{j=1}^m g_{ij}(X_{n+\frac{1}{2}}^a) \Delta w_{j,n+\frac{1}{2}}^\ell \\ &\quad + \sum_{j,k=1}^m h_{ijk}(X_{n+\frac{1}{2}}^a) (\Delta w_{j,n+\frac{1}{2}}^\ell \Delta w_{k,n+\frac{1}{2}}^\ell - \Omega_{jk} \Delta t_{\ell-1}/2). \end{aligned} \quad (42)$$

It can be shown that [22]

Lemma 6.3. *For all integers $p \geq 2$, there exists a constant K_p such that*

$$\mathbb{E} \left[\max_{0 \leq n \leq N} \|X_n^f - X_n^a\|^p \right] \leq K_p \Delta t^{p/2}.$$

Let's denote the average fine and antithetic path as follows

$$\overline{X}_n^f \equiv \frac{1}{2}(X_n^f + X_n^a).$$

The main results of [22] is the following theorem:

Theorem 6.4. *For all $p \geq 2$, there exists a constant K_p such that*

$$\mathbb{E} \left[\max_{0 \leq n \leq N} \|\overline{X}_n^f - X_n^c\|^p \right] \leq K_p \Delta t^p.$$

This together with a classical strong convergence result for Milstein discretization allows to estimate the MLMC variance for smooth payoffs. In the case of payoff which is a smooth function of the final state $x(T)$, taking $p=2$ in Lemma 6.1, $p=4$ in Lemma 6.3 and $p=2$ in Theorem 6.4, immediately gives the result that the multilevel variance

$$\mathbb{V} \left[\frac{1}{2} \left(P(X_N^f) + P(X_N^a) \right) - P(X_N^c) \right]$$

has an $O(\Delta t^2)$ upper bound. This matches the convergence rate for the multilevel method for scalar SDEs using the standard first order Milstein discretisation, and is much better than the $O(\Delta t)$ convergence obtained with the Euler-Maruyama discretisation.

However, very few financial payoff functions are twice differentiable on the entire domain \mathbb{R}^d . A more typical 2D example is a call option based on the minimum of two assets,

$$P(x(T)) \equiv \max(0, \min(x_1(T), x_2(T)) - K),$$

which is piecewise linear, with a discontinuity in the gradient along the three lines (s, K) , (K, s) and (s, s) for $s \geq K$.

To handle such payoffs, an assumption which bounds the probability of the solution of the SDE having a value at time T close to such lines with discontinuous gradients is needed.

Assumption 6.5. *The payoff function $P \in C(\mathbb{R}^d, \mathbb{R})$ has a uniform Lipschitz bound, so that there exists a constant L such that*

$$|P(x) - P(y)| \leq L |x - y|, \quad \forall x, y \in \mathbb{R}^d,$$

and the first and second derivatives exist, are continuous and have uniform bound L at all points $x \notin K$, where K is a set of zero measure, and there exists a constant c such that the probability of the SDE solution $x(T)$ being within a neighbourhood of the set K has the bound

$$\mathbb{P} \left(\min_{y \in K} \|x(T) - y\| \leq \varepsilon \right) \leq c \varepsilon, \quad \forall \varepsilon > 0.$$

In a 1D context, Assumption 6.5 corresponds to an assumption of a locally bounded density for $x(T)$.

Giles and Szpruch in [22] proved the following result

Theorem 6.6. *If the payoff satisfies Assumption 6.5, then*

$$\mathbb{E} \left[\left(\frac{1}{2} (P(X_N^f) + P(X_N^a)) - P(X_N^c) \right)^2 \right] = o(\Delta t^{3/2-\delta})$$

for any $\delta > 0$.

6.4 Piecewise linear interpolation analysis

The piecewise linear interpolant $X^c(t)$ for the coarse path is defined within the coarse timestep interval $[t_k, t_{k+1}]$ as

$$X^c(t) \equiv (1-\lambda) X_k^c + \lambda X_{k+1}^c, \quad \lambda \equiv \frac{t - t_k}{t_{k+1} - t_k}.$$

Likewise, the piecewise linear interpolants $X^f(t)$ and $X^a(t)$ are defined on the fine timestep $[t_k, t_{k+\frac{1}{2}}]$ as

$$X^f(t) \equiv (1-\lambda) X_k^f + \lambda X_{k+\frac{1}{2}}^f, \quad X^a(t) \equiv (1-\lambda) X_k^a + \lambda X_{k+\frac{1}{2}}^a, \quad \lambda \equiv \frac{t - t_k}{t_{k+\frac{1}{2}} - t_k},$$

and there is a corresponding definition for the fine timestep $[t_{k+\frac{1}{2}}, t_{k+1}]$. It can be shown that [22]

Theorem 6.7. *For all $p \geq 2$, there exists a constant K_p such that*

$$\sup_{0 \leq t \leq T} \mathbb{E} [\|X^f(t) - X^a(t)\|^p] \leq K_p \Delta t^{p/2},$$

$$\sup_{0 \leq t \leq T} \mathbb{E} [\|\overline{X}^f(t) - X^c(t)\|^p] \leq K_p \Delta t^p,$$

where $\overline{X}^f(t)$ is the average of the piecewise linear interpolants $X^f(t)$ and $X^a(t)$.

For an Asian option, the payoff depends on the average

$$x_{ave} \equiv T^{-1} \int_0^T x(t) dt.$$

This can be approximated by integrating the appropriate piecewise linear interpolant which gives

$$X_{ave}^c \equiv T^{-1} \int_0^T X^c(t) dt = N^{-1} \sum_{n=0}^{N-1} \frac{1}{2} (X_n^c + X_{n+1}^c),$$

$$X_{ave}^f \equiv T^{-1} \int_0^T X^f(t) dt = N^{-1} \sum_{n=0}^{N-1} \frac{1}{4} (X_n^f + 2X_{n+\frac{1}{2}}^f + X_{n+1}^f),$$

$$X_{ave}^a \equiv T^{-1} \int_0^T X^a(t) dt = N^{-1} \sum_{n=0}^{N-1} \frac{1}{4} (X_n^a + 2X_{n+\frac{1}{2}}^a + X_{n+1}^a).$$

Due to Hölder's inequality,

$$\begin{aligned} \mathbb{E} [\|X_{ave}^f - X_{ave}^a\|^p] &\leq T^{-1} \int_0^T \mathbb{E} [\|X^f(t) - X^a(t)\|^p] dt \\ &\leq \sup_{[0,T]} \mathbb{E} [\|X^f(t) - X^a(t)\|^p], \end{aligned}$$

and similarly

$$\mathbb{E} [\|\frac{1}{2}(X_{ave}^f + X_{ave}^a) - X_{ave}^c\|^p] \leq \sup_{[0,T]} \mathbb{E} [\|\overline{X}^f(t) - X^c(t)\|^p],$$

Hence, if the Asian payoff is a smooth function of the average, then we obtain a second order bound for the multilevel correction variance.

This analysis can be extended to include payoffs which are a smooth function of a number of intermediate variables, each of which is a linear functional of the path $x(t)$ of the form

$$\int_0^T g^T(t) x(t) \mu(dt),$$

for some vector function $g(t)$ and measure $\mu(dt)$. This includes weighted averages of $x(t)$ at a number of discrete times, as well as continuously-weighted averages over the whole time interval.

As with the European options, the analysis can also be extended to payoffs which are Lipschitz functions of the average, and have first and second derivatives which exist, and are continuous and uniformly bounded, except for a set of points K of zero measure.

Assumption 6.8. *The payoff $P \in C(\mathbb{R}^d, \mathbb{R})$ has a uniform Lipschitz bound, so that there exists a constant L such that*

$$|P(x) - P(y)| \leq L |x - y|, \quad \forall x, y \in \mathbb{R}^d,$$

and the first and second derivatives exist, are continuous and have uniform bound L at all points $x \notin K$, where K is a set of zero measure, and there exists a constant c such that the probability of x_{ave} being within a neighbourhood of the set K has the bound

$$\mathbb{P} \left(\min_{y \in K} \|x_{ave} - y\| \leq \varepsilon \right) \leq c \varepsilon, \quad \forall \varepsilon > 0.$$

Theorem 6.9. *If the payoff satisfies Assumption 6.8, then*

$$\mathbb{E} \left[\left(\frac{1}{2} (P(X_{ave}^f) + P(X_{ave}^a)) - P(X_{ave}^c) \right)^2 \right] = o(\Delta t^{3/2-\delta})$$

for any $\delta > 0$.

We refer the reader to [22] for more details.

6.5 Simulations for antithetic Monte Carlo

Here we present numerical simulations for a European option for process simulated by X^f , X^a and X^c defined in section 6.2 with initial conditions $x_1(0) = x_2(0) = 1$. The results in Figure 2 are for a European call option with terminal time 1 and strike $K = 1$, that is $P = (x(T) - K)^+$. The top left plot shows the behaviour of the variance of both P_ℓ and $P_\ell - P_{\ell-1}$. The superimposed reference slope with rate 1.5 indicates that the variance $\mathbb{V}_\ell = \mathbb{V}[P_\ell - P_{\ell-1}] = O(\Delta t_\ell^{1.5})$, corresponding to $O(\epsilon^{-2})$ computational complexity of antithetic MLMC. The top right plot shows that $\mathbb{E}[P_\ell - P_{\ell-1}] = O(\Delta t_\ell)$. The bottom left plot shows the computational complexity C (as defined in Theorem 2.1) with desired accuracy ϵ . The plot is of $\epsilon^2 C$ versus ϵ , because we expect to see that $\epsilon^2 C$ is only weakly dependent on ϵ for MLMC. For standard Monte Carlo, theory predicts that $\epsilon^2 C$ should be proportional to the number of timesteps on the finest level, which in turn is roughly proportional to ϵ^{-1} due to the weak convergence order. For accuracy $\epsilon = 10^{-4}$, the antithetic MLMC is approximately 500 times more efficient than the standard Monte Carlo. The bottom right plot shows that $\mathbb{V}[X_{1,\ell} - X_{1,\ell-1}] = O(\Delta t_\ell)$. This corresponds to standard strong convergence

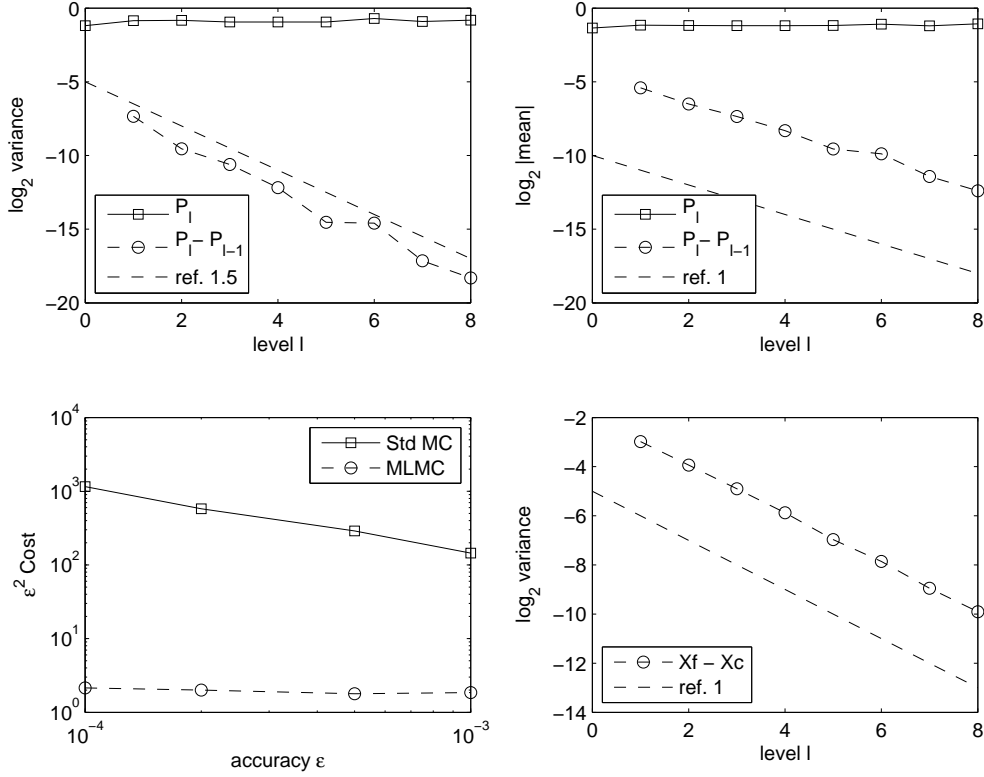


Figure 2: Call option.

of order 0.5. We have also tested the algorithm presented in [22] for approximation of Asian options. Our results were almost identical as for European options. In order to treat the lookback, digital and barrier options we found that a suitable antithetic approximation to the Lévy areas are needed. For suitable modification of the antithetic MLMC estimator we performed numerical experiments where we obtained $O(\epsilon^{-2} \log(\epsilon)^2)$ complexity for estimating barrier, digital and lookback options. Currently, we are working on theoretical justification of our results.

7 Other uses of multilevel method

7.1 SPDEs

Multilevel method has been used for a number of parabolic and elliptic SPDE applications [4, 10, 27] but the first use for a financial SPDE is in a new paper by Giles & Reisinger [21].

This paper considers an unusual SPDE which results from modelling credit default probabilities,

$$\Delta p = -\mu \frac{\partial p}{\partial x} \Delta t + \frac{1}{2} \frac{\partial^2 p}{\partial x^2} \Delta t - \sqrt{\rho} \frac{\partial p}{\partial x} \Delta M_t, \quad x > 0 \quad (43)$$

subject to boundary condition $p(0, t) = 0$. Here $p(x, t)$ represents the probability density function for firms being a distance x from default at time t . The diffusive term is due to idiosyncratic factors affecting individual firms, while the stochastic term due to the scalar Brownian motion M_t corresponds to the systemic movement due to random market effects affecting all firms.

Using a Milstein time discretisation with uniform timestep k , and a central space discretisation of the spatial derivatives with uniform spacing h gives the numerical approximation

$$\begin{aligned} p_j^{n+1} = & p_j^n - \frac{\mu k + \sqrt{\rho k} Z_n}{2h} (p_{j+1}^n - p_{j-1}^n) \\ & + \frac{(1-\rho)k + \rho k Z_n^2}{2h^2} (p_{j+1}^n - 2p_j^n + p_{j-1}^n), \end{aligned} \quad (44)$$

where the Z_n are standard Normal random variables so that $\sqrt{h}Z_n$ corresponds to an increment of the driving scalar Brownian motion.

The paper shows that the requirement for mean-square stability as the grid is refined and $k, h \rightarrow 0$ is $k/h^2 \leq (1 + 2\rho^2)^{-1}$, and in addition the accuracy is $O(k, h^2)$. Because of this, the multilevel treatment considers a sequence of grids with $h_\ell = 2h_{\ell-1}$, $k_\ell = 4k_{\ell-1}$.

The multilevel implementation is very straightforward, with the Brownian increments for the fine path being summed pairwise to give the corresponding Brownian increments for the coarse path. The payoff corresponds to different tranches of a credit derivative that depends on a numerical approximation of the integral

$$\int_0^\infty p(x, t) \Delta x.$$

The computational cost increases by factor 8 on each level, and numerical experiments indicate that the variance decreases by factor 16. The MLMC Theorem still applies in this case, with $\beta = 4$ and $\gamma = 3$, and so the overall computational complexity to achieve an $O(\epsilon)$ RMS error is again $\mathcal{O}(\epsilon^{-2})$.

7.2 Nested simulation

The pricing of American options is one of the big challenges for Monte Carlo methods in computational finance, and Belomestny & Schoenmakers have recently written a very interesting paper on the use of multilevel Monte Carlo for this purpose [5]. Their method is based on Anderson & Broadie's dual simulation method [1] in which a key component at each timestep in the simulation is to estimate a conditional expectation using a number of sub-paths.

In their multilevel treatment, Belomestny & Schoenmakers use the same uniform timestep on all levels of the simulation. The quantity which changes between different levels of simulation is the number of sub-samples used to estimate the conditional expectation.

To couple the coarse and fine levels, the fine level uses N_ℓ sub-samples, and the coarse level uses $N_{\ell-1} = N_\ell/2$ of them. Similar research by N. Chen¹ found the multilevel correction variance is reduced if the payoff on the coarse level is replaced by an average of the payoffs obtained using the first $N_\ell/2$ and the second $N_\ell/2$ samples. This is similar in some ways to the antithetic approach described in section 6.

In future research, Belomestny & Schoenmakers intend to also change the number of timesteps on each level, to increase the overall computational benefits of the multilevel approach.

7.3 Truncated series expansions

Building on earlier work by Broadie and Kaya [7], Glasserman and Kim have recently developed an efficient method [25] of exactly simulating the Heston stochastic volatility model [29].

The key to their algorithm is a method of representing the integrated volatility over a time interval $[0, T]$, conditional on the initial and final values, v_0 and v_T as

$$\left(\int_0^T V_s ds \mid V_0 = v_0, V_T = v_T \right) \stackrel{d}{=} \sum_{n=1}^{\infty} x_n + \sum_{n=1}^{\infty} y_n + \sum_{n=1}^{\infty} z_n$$

where x_n, y_n, z_n are independent random variables.

In practice, they truncate the series expansions at a level which ensures the desired accuracy, but a more severe truncation would lead to a tradeoff between accuracy and computational cost. This makes the algorithm a candidate for a multilevel treatment in which level ℓ computation performs the truncation at N_ℓ (taken to be the same for all three series, for simplicity).

To give more details, the level ℓ computation would use

$$\sum_{n=1}^{N_\ell} x_n + \sum_{n=1}^{N_\ell} y_n + \sum_{n=1}^{N_\ell} z_n$$

while the level $\ell-1$ computation would use

$$\sum_{n=1}^{N_{\ell-1}} x_n + \sum_{n=1}^{N_{\ell-1}} y_n + \sum_{n=1}^{N_{\ell-1}} z_n$$

with the same random variables x_n, y_n, z_n .

This kind of multilevel treatment has not been tested experimentally, but it seems that it might yield some computational savings even though Glasserman and Kim typically retain only 10 terms in their summations through the use of a carefully constructed estimator for the truncated remainder. In other circumstances requiring more terms to be retained, the savings may be larger.

¹unpublished, but presented at the MCQMC12 conference.

7.4 Mixed precision arithmetic

The final example of the use of multilevel is unusual, because it concerns the computer implementation of Monte Carlo algorithms.

In the latest CPUs from Intel and AMD, each core has a vector unit which can perform 8 single precision or 4 double precision operations with one instruction. Together with the obvious fact that double precision variables are twice as big as single precision variables and so require twice as much time to transfer, in bulk, it leads to single precision computations being twice as fast as double precision computations. On GPUs (graphics processors) the difference in performance can be even larger, up to a factor of eight in the most extreme cases.

This raises the question of whether single precision arithmetic is sufficient for Monte Carlo simulation. In general, our view is that the errors due to single precision arithmetic are much smaller than the errors due to

- statistical error due to Monte Carlo sampling;
- bias due to SDE discretisation;
- model uncertainty.

We have just two concerns with single precision accuracy:

- there can be significant errors when averaging the payoffs unless one uses binary tree summation [?] to perform the summation;
- when computing Greeks using “bumping”, the single precision inaccuracy can be greatly amplified if a small bump is used.

Our advice would be to always use double precision for the final accumulation of payoff values, and pathwise sensitivity analysis as much as possible for computing Greeks, but if there remains a need for the path simulation to be performed in double precision then one could use two-level approach in which level 0 corresponds to single precision and level 1 corresponds to double precision.

On both levels one would use the same random numbers. The multilevel analysis would then give the optimal allocation of effort between the single precision and double precision computations. Since it is likely that most of the calculations would be single precision, the computational savings would be a factor two or more compared to standard double precision calculations.

8 Multilevel Quasi-Monte Carlo

In Theorem 2.1, if $\beta > \gamma$, so that rate at which the multilevel variance decays with increasing grid level is greater than the rate at which the computational

cost increases, then the dominant computational cost is on the coarsest levels of approximation.

Since coarse levels of approximation correspond to low-dimensional numerical quadrature, it is quite natural to consider the use of quasi-Monte Carlo techniques. This has been investigated by Giles & Waterhouse [23] in the context of scalar SDEs with a Lipschitz payoff. Using the Milstein approximation with a doubling of the number of timesteps on each level gives $\beta = 2$ and $\gamma = 1$. They used a rank-1 lattice rule to generate the quasi-random numbers, randomisation with 32 independent offsets to obtain confidence intervals, and a standard Brownian Bridge construction of the increments of the driving Brownian process.

Their empirical observation was that MLMC on its own was better than QMC on its own, but the combination of the two was even better. The QMC treatment greatly reduced the variance per sample for the coarsest levels, resulting in significantly reduced costs overall. In the simplest case of a European call option, shown in Figure 3, the top left plot shows the reduction in the variance per sample as the number of QMC points is increased. The benefit is

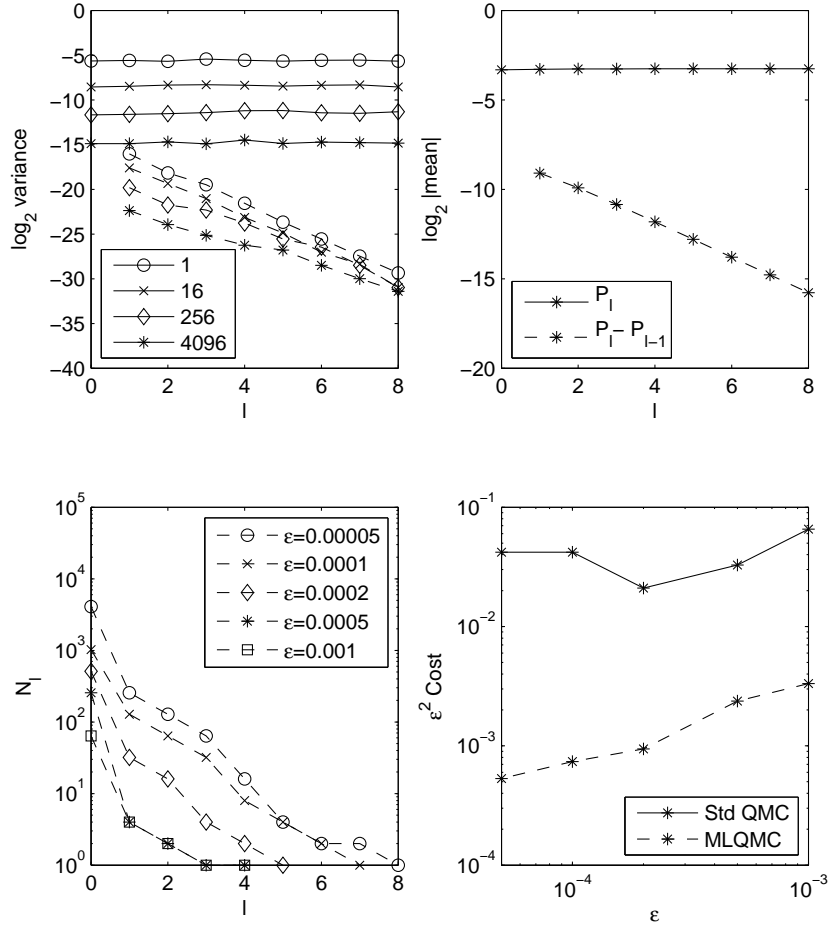


Figure 3: European call option (Fig 6.1 from [23])

much greater on the coarsest levels than on the finest levels. In the bottom two plots, the number of QMC points on each level is determined automatically to obtain the required accuracy; see [23] for the precise details. Overall, the computational complexity appears to be reduced from $O(\epsilon^{-2})$ to approximately $O(\epsilon^{-1.5})$.

Giles & Waterhouse interpreted the fact that the variance is not reduced on the finest levels as being due to a lack of significant low-dimensional content. i.e. the difference in the two payoffs due to neighbouring grid levels is due to the difference in resolution of the driving Brownian path, and this is inherently of high dimensionality. This suggests that in other applications with $\beta < \gamma$, which would lead to the dominant cost being on the finest levels, then the use of quasi-Monte Carlo methods is unlikely to yield any benefits.

Further research is needed in this area to investigate the use of other low-discrepancy sequences (e.g. Sobol) and other ways of generating the Brownian increments (e.g. PCA). We also refer the reader to [15] for some results for randomized multilevel quasi-Monte Carlo.

9 Conclusions

In the past 6 years, considerable progress has been achieved with the multilevel Monte Carlo method for financial options based on underlying assets described by Brownian diffusions, jump diffusions, and more general Lévy processes.

The multilevel approach is conceptually very simple. In essence it is a recursive control variate strategy, using a coarse path simulation as a control variate for a fine path simulation, relying on strong convergence properties to ensure a very strong correlation between the two.

In practice, the challenge is to couple the coarse and fine path simulations as tightly as possible, minimising the difference in the payoffs obtained for each. In doing this, there is considerable freedom to be creative, as shown in the use of Brownian Bridge constructions to improve the variance for lookback and barrier options, and in the antithetic estimators for multi-dimensional SDEs which would require the simulation of Lévy areas to achieve first order strong convergence. Another challenge is avoiding large payoff differences due to discontinuous payoffs; here one can often use either conditional expectations to smooth the payoff, or a change of measure to ensure that the coarse and fine paths are on the same side of the discontinuity.

Overall, multilevel methods are being used for an increasingly wide range of applications. The biggest savings are in situations in which the coarsest approximation is very much cheaper than the finest. If the finest level of approximation has only 32 timesteps, then there are very limited savings to be achieved, but if the finest level has 256 timesteps, then the potential savings are much larger.

Looking to the future, exciting areas for further research include:

- more research on multilevel techniques for American and Bermudan options;
- more investigation of multilevel Quasi Monte Carlo methods;
- use of multilevel ideas for completely new financial applications, such as Gaussian copula and new SPDE models.

References

- [1] L. Andersen and M. Broadie. A primal-dual simulation algorithm for pricing multi-dimensional American options. *Management Science*, 50(9):1222–1234, 2004.
- [2] A. Asmussen and P. Glynn. *Stochastic Simulation*. Springer, New York, 2007.
- [3] R. Avikainen. On irregular functionals of SDEs and the Euler scheme. *Finance and Stochastics*, 13(3):381–401, 2009.
- [4] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numerische Mathematik*, 119(1):123–161, 2011.
- [5] D. Belomestny and J. Schoenmakers. Multilevel dual approach for pricing American style derivatives. Preprint 1647, WIAS, 2011.
- [6] M. Broadie, P. Glasserman, and S. Kou. A continuity correction for discrete barrier options. *Mathematical Finance*, 7(4):325–348, 1997.
- [7] M. Broadie and O. Kaya. Exact simulation of stochastic volatility and other affine jump diffusion processes. 54(2):217–231, 2006.
- [8] S. Burgos and M.B. Giles. Computing Greeks using multilevel path simulation. In L. Plaskota and H. Woźniakowski, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*. Springer-Verlag, 2012.
- [9] J.M.C. Clark and R.J. Cameron. The maximum rate of convergence of discrete approximations for stochastic differential equations. In B. Grigelionis, editor, *Stochastic Differential Equations, no. 25 in Lecture Notes in Control and Information Sciences*. Springer-Verlag, 1980.
- [10] K.A. Cliffe, M.B. Giles, R. Scheichl, and A. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.
- [11] K. Debrabant, M. B. Giles, and A. Rossler. Numerical analysis of multilevel monte carlo path simulation using milstein discretization: scalar case. *Technical report*, 2011.
- [12] S. Dereich. Multilevel Monte Carlo algorithms for Lévy-driven SDEs with Gaussian correction. *Annals of Applied Probability*, 21(1):283–311, 2011.
- [13] S. Dereich and F. Heidenreich. A multilevel monte carlo algorithm for lévy-driven stochastic differential equations. *Stochastic Processes and their Applications*, 2011.

- [14] J.G. Gaines and T.J. Lyons. Random generation of stochastic integrals. *SIAM Journal of Applied Mathematics*, 54(4):1132–1146, 1994.
- [15] Thomas Gerstner and Marco Noll. Some results for randomized multilevel quasi-Monte Carlo path simulation. *Chapter in this book*, 2012.
- [16] M.B. Giles. Monte Carlo evaluation of sensitivities in computational finance. Technical Report NA07/12, 2007.
- [17] M.B. Giles. Improved multilevel Monte Carlo convergence using the Milstein scheme. In A. Keller, S. Heinrich, and H. Niederreiter, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 343–358. Springer-Verlag, 2008.
- [18] M.B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.
- [19] M.B. Giles. Multilevel Monte Carlo for basket options. In *Proceedings of the Winter Simulation Conference 2009*, 2009.
- [20] M.B. Giles, D.J. Higham, and X. Mao. Analysing multilevel Monte Carlo for options with non-globally Lipschitz payoff. *Finance and Stochastics*, 13(3):403–413, 2009.
- [21] M.B. Giles and C. Reisinger. Stochastic finite differences and multilevel Monte Carlo for a class of SPDEs in finance. *SIAM Journal of Financial Mathematics*, to appear, 2012.
- [22] M.B. Giles and L. Szpruch. Antithetic Multilevel Monte Carlo estimation for multi-dimensional SDEs without Lévy area simulation. *Arxiv preprint arXiv:1202.6283*, 2012.
- [23] M.B. Giles and B.J. Waterhouse. Multilevel quasi-Monte Carlo path simulation. In *Advanced Financial Modelling*, Radon Series on Computational and Applied Mathematics, pages 165–181. de Gruyter, 2009.
- [24] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, New York, 2004.
- [25] P. Glasserman and K.-K. Kim. Gamma expansion of the Heston stochastic volatility model. *Finance and Stochastics*, 15(2):267–296, 2011.
- [26] P. Glasserman and N. Merener. Convergence of a discretization scheme for jump-diffusion processes with state-dependent intensities. *Proc. Royal Soc. London A*, 460:111–127, 2004.
- [27] S. Graubner. Multi-level Monte Carlo Methoden für stochastische partial Differentialgleichungen. Diplomarbeit, TU Darmstadt, 2008.

- [28] S. Heinrich. *Multilevel Monte Carlo Methods*, volume 2179 of *Lecture Notes in Computer Science*, pages 58–67. Springer-Verlag, 2001.
- [29] S.I. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6:327–343, 1993.
- [30] I. Karatzas and S.E. Shreve. *Brownian Motion and Stochastic Calculus*. Graduate Texts in Mathematics, Vol. 113. Springer, New York, 1991.
- [31] A. Kebaier. Statistical Romberg extrapolation: a new variance reduction method and applications to options pricing. *Annals of Applied Probability*, 14(4):2681–2705, 2005.
- [32] P. Kloeden and A. Neuenkirch. Convergence of numerical methods for stochastic differential equations in mathematical finance. *Chapter in this book*, 2012.
- [33] P.E. Kloeden, A. Neuenkirch, and R. Pavani. Multilevel monte carlo for stochastic differential equations with additive fractional noise. *Annals of Operations Research*, pages 1–22, 2011.
- [34] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, Berlin, 1992.
- [35] X. Mao and L. Szpruch. Strong convergence rates for backward euler-maruyama method for non-linear dissipative-type stochastic differential equations with super-linear diffusion coefficients. *Stochastics*, 2012.
- [36] R.C. Merton. Option pricing when underlying stock returns are discontinuous. *Journal of Finance*, 3:125–144, 1976.
- [37] T. Müller-Gronbach. The optimal uniform approximation of systems of stochastic differential equations. *The Annals of Applied Probability*, 12(2):664–690, 2002.
- [38] T. Müller-Gronbach. *Strong approximation of systems of stochastic differential equations*. Habilitation thesis, TU Darmstadt, 2002.
- [39] G. Pagès. Multi-step Richardson-Romberg extrapolation: remarks on variance control and complexity. *Monte Carlo Methods and Applications*, 13(1):37–70, 2007.
- [40] E. Platen and N. Bruti-Liberati. *Numerical Solution of Stochastic Differential Equations With Jumps in Finance*. Springer, 2010.
- [41] T. Rydén and M. Wiktorsson. On the simulation of iterated Itô integrals. *Stochastic Processes and their Applications*, 91(1):151–168, 2001.

- [42] A.L. Speight. A multilevel approach to control variates. *Journal of Computational Finance*, 12:1–25, 2009.
- [43] A.L. Speight. Multigrid techniques in economics. *Operations Research*, 58(4):1057–1078, 2010.
- [44] L. Szpruch, X. Mao, D.J. Higham, and J. Pan. Numerical simulation of a strongly nonlinear ait-sahalia-type interest rate model. *BIT Numerical Mathematics*, 51(2):405–425, 2011.
- [45] M. Wiktorsson. Joint characteristic function and simultaneous simulation of iterated Itô integrals for multiple independent Brownian motions. *Annals of Applied Probability*, 11(2):470–487, 2001.
- [46] Y. Xia. Multilevel Monte Carlo method for jump-diffusion SDEs. *Arxiv preprint arXiv:1106.4730*, 2011.
- [47] Y. Xia and M.B. Giles. Multilevel path simulation for jump-diffusion SDEs. In L. Plaskota and H. Woźniakowski, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*. Springer-Verlag, 2012.